

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»**

**Факультет інформатики та обчислювальної техніки
Кафедра автоматики та управління в технічних системах**

До захисту допущено:

Завідувач кафедри

_____ Олександр РОЛІК

«__» _____ 20__ р.

**Дипломний проєкт
на здобуття ступеня бакалавра
за освітньо-професійною програмою «Програмне забезпечення
інформаційно-комунікаційних систем»
спеціальності 121 «Інженерія програмного забезпечення»
на тему: «Автоматизована система зворотного зв'язку з клієнтами»**

Виконав (-ла):

студент (-ка) IV курсу, групи ІТ-61

Нагуляк Андрій Сергійович

Керівник:

Асистент кафедри АУТС

Шинкевич Микола Костянтинович

Рецензент:

Засвідчую, що у цьому дипломному проєкті
немає запозичень з праць інших авторів без
відповідних посилань.

Студент (-ка) _____

Київ – 2020 року

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра автоматики та управління в технічних системах

Рівень вищої освіти – перший (бакалаврський)

Спеціальність – 121 «Інженерія програмного забезпечення»

Освітньо-професійна програма «Програмне забезпечення інформаційно-комунікаційних систем»

ЗАТВЕРДЖУЮ

Завідувач кафедри

_____ Олександр РОЛІК

«__» _____ 20__ р.

ЗАВДАННЯ
на дипломний проєкт студенту
Нагуляку Андрію Сергійовичу

1. Тема проєкту «Автоматизована система зворотного зв'язку з клієнтами», керівник проєкту асистент Шинкевич Микола Константинович, затверджені наказом по університету від 07 травня 2020 р. №1081-с

2. Термін подання студентом проєкту 09.06.2020

3. Вихідні дані до проєкту

Мова програмування TypeScript, фреймворки Angular 2, ASP .Net core, середовище програмування Visual Code, Visual Studio, СУБД – MSSQL.

4. Зміст пояснювальної записки

1. Вступ 2. Огляд існуючих рішень 3. Аналіз особливостей системи зворотного зв'язку 4. Сценарії використання 5. Розробка програмного забезпечення 6. Тестування системи зворотного зв'язку 7. Впровадження та використання системи

5. Перелік графічного матеріалу (із зазначенням обов'язкових креслеників, плакатів, презентацій тощо)

Діаграма варіантів використання, діаграма компонентів, діаграма процесу, діаграма структури бази даних

6. Дата видачі завдання _____

Календарний план

№ з/п	Назва етапів виконання дипломного проекту	Термін виконання етапів проекту	Примітка
1	Вибір тематичного напрямку та узгодження теми дипломного проекту	22.02.2020	
2	Аналіз методів та засобів зворотного зв'язку між клієнтом та компанією	15.04.2020	
3	Створення сценаріїв використання системи та ролевої моделі	24.04.2020	
4	Вибір засобів реалізації системи	27.04.2020	
5	Розробка прототипу системи	06.05.2020	
6	Реалізація системи	20.05.2020	
7	Тестування системи	23.05.2020	
8	Оформлення пояснювальної записки	03.06.2020	
9	Передзахист дипломного проекту	04.06.2020	

Студент

Андрій НАГУЛЯК

Керівник

Микола ШИНКЕВИЧ

АНОТАЦІЯ

Нагуляк А.С. Автоматизована система зворотного зв'язку з клієнтами. КПІ ім. Ігоря Сікорського, Київ, 2020.

Ключові слова: Typescript, C#, Angular. ASP .Net core, система зворотного зв'язку, діаграма використання.

Основна частина документу викладена у пояснювальній записці, виконаній на 84 сторінках, та містить 34 рисунків та 79 таблиць.

Об'єктом розробки є система зворотного зв'язку між клієнтом та компанією. Мета розробки – створення автоматизованої системи комунікації між компанією та споживачем .

У дипломному проєкті проведено аналіз та систематизація існуючих модулів зворотного зв'язку та відповідно до результатів розроблено автоматизовану систему зворотного зв'язку між споживачами та компаніями. Система є реалізованою на базі клієнт-сервісної архітектури, та покрита всіма можливими тестами для коректного виконання сценаріїв використання.

Отримані результати можуть бути корисними при створенні подібних систем.

SUMMARY

Nagulyak A. S. Automated customer feedback system. Igor Sikorsky KPI, Kyiv, 2020.

Keywords: Typescript, C #, Angular. ASP .Net core, feedback system, usage diagram.

The bulk of the document is outlined in the explanatory note, with 84 pages, and contains 34 figures and 79 tables.

The object of development is a feedback system between the client and the company.

The purpose of the development is to create an automated communication system between the company and the consumer. The graduation project has a thorough analysis of the existing decisions, based on the existing feedback modules, and in accordance with the results developed an automated feedback system between consumers and companies. The system is implemented based on client-service architecture, and is covered by all possible tests for the correct execution of usage scenarios.

The results obtained can be useful in creating such systems

**Пояснювальна записка
до дипломного проєкту
на тему: «Автоматизована система зворотного
зв'язку з клієнтами»**

Київ – 2020 року

ЗМІСТ

ВСТУП.....	10
1 ОГЛЯД ІСНУЮЧИХ РІШЕНЬ.....	13
1.1 Існуючі модулі зворотного зв'язку.....	13
1.2 Обґрунтування доцільності розробки	15
Висновки до розділу.....	16
2 АНАЛІЗ ОСОБЛИВОСТЕЙ СИСТЕМИ ЗВОРОТНОГО ЗВ'ЯЗКУ	17
2.1 Опис та аналіз предметного середовища	17
2.2 Функціональні вимоги до системи	18
2.3 Нефункціональні вимоги до системи	21
Висновки до розділу.....	22
3 СЦЕНАРІЙ ВИКОРИСТАННЯ СИСТЕМИ.....	23
3.1 Сценарії використання системи.....	23
Висновки до розділу.....	45
4 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.....	46
4.1 Вибір технологій розробки	46
4.2 Розробка прототипу системи	47
4.3 Розробка структурної схеми бази даних	54
4.4 Структура проекту.....	57
4.4.1 Структура серверної частини	57
4.4.2 Структура клієнтської частини	59
4.4.3 Структура тестової частини	59

					ІТ61.140БАК.004 ПЗ				
Змн.	Арк.	№ докум.	Підпис	Дата					
Розроб.		Нагуляк А.С.			Система автоматизованого зворотного зв'язку з клієнтами. Пояснювальна записка		Літ.	Арк.	Аркушів
Перевір.		Шинкевич М.К.						2	89
Реценз.							КПІ ім. Ігоря Сікорського ФІОТ, гр. ІТ-61		
Н. Контр.		Шинкевич М.К.							
Затверд.									

4.5 Реалізація системи	60
4.5.1 Автентифікація	60
4.5.2 Адміністрування компанії.	63
4.5.3 Відгуки.....	77
Висновки до розділу.....	82
5 ТЕСТУВАННЯ СИСТЕМИ ЗВОРОТНОГО ЗВ'ЯЗКУ	83
5.1 Доцільність тестування	83
5.2 Unit-тестування	83
5.3 Інтеграційне тестування	85
5.4 End-to-end тестування	88
Висновки до розділу.....	90
6 ВПРОВАДЖЕННЯ ТА ВИКОРИСТАННЯ СИСТЕМИ.....	91
6.1 Апаратні вимоги для використання системи	91
6.2 Інструкція з встановлення для розробки	91
6.3 Висновки до розділу	92
ВИСНОВКИ	93
ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ	94

ВСТУП

Успіх організації напряму залежить від задоволення клієнта, тому особливо важливим є налаштування зручного зв'язку споживача з компанією для отримання інформації, яка може знадобитися для виявлення можливих недоліків та забезпечення покращень в роботі організації. Також за статистикою, компанії які цінують думку своїх клієнтів та мають відкриту комунікацію з ними, приваблюють цим самим більше споживачів. Саме тому питання зв'язку між клієнтом та компанією є одним з найголовніших факторів успішності організації.

Кращими стають ті, що постійно шукають способи зробити себе ще кращими. Термін «зворотний зв'язок» використовується для опису корисної інформації чи критики попередніх дії або поведінки людини, переданих іншій людині (або групі), які можуть використовувати цю інформацію для коригування та поліпшення дій або поведінки. Така інформація дуже корисна і буде використовуватися для прийняття важливих рішень. Для провідних компаній «постійне поліпшення» - це не просто слова. Вони враховують всі відгуки зацікавлених сторін - клієнтів, співробітників, постачальників та інших. Найкращі компанії не тільки добре сприймають відгуки, але і навмисно запитують їх. Вони знають, що зворотний зв'язок корисний тільки тоді, коли він підкреслює як слабкі, так і сильні сторони.

"Найглибший принцип людської природи - це тяга, бути оціненим". Вільям Джеймс. [1]

Зворотний зв'язок дає змогу по-різному дивитись на ситуації та на себе, відкриває саморефлексію та ворота для змін. Він може допомогти в розробленні продуктів та послуг, забезпечуючи можливість компанії зосередити свої ресурси на реалізації потреб споживачів. Аналіз очікувань, вміння не просто збирати критику, а й ефективно нейтралізувати її, сприятимуть досягнення адекватних взаємин з клієнтами. Люди, які скаржаться на ваші продукти і послуги насправді допомагають вам краще працювати.

					ІТ61.140БАК.004 ПЗ	Арк.
						10
Змн.	Арк.	№ докум.	Підпис	Дата		

Зважаючи на сьогоdnішній розвиток Інтернету, практично вся комунікація проходить через нього. Проаналізувавши ситуацію з зворотним зв'язком в сучасному Інтернеті, я виявив декілька проблем.

По-перше не всі компанії мають свої web-сайти. Зокрема новоутворені не мають змоги та ресурсів створити свій власний сайт, а тому зв'язатися з ними чи залишити відгук в мережі Інтернет дуже складно. Існують системи які призначенні для оцінки компанії чи залишення відгуку, проте і ці системи не надають змогу в повній мірі контактувати з компанією напpяму. Спираючись на це, малі компанії не мають можливості зареєструвати свій повноцінний модуль зворотного зв'язку, а лише переглядати та відповідати на відгуки.

По-друге, не всі сайти мають модулі зворотного зв'язку. Це пов'язано з застарілістю деяких сайтів або недоцільністю використання зворотного зв'язку на «легких» сайтах, де модуль тільки ускладнить інтерфейс.

По-третє, компанії які мають модулі зворотного зв'язку, зазвичай створюють його не в значній мірі. Тобто це може бути лише блок з гарячими лініями чи форма для надання відгуку, які не дають змоги повноцінного зв'язку з клієнтом.

Актуальність теми дипломного проєкту полягає у тому, щоб вирішити проблему з комунікацією між компаніями та клієнтами, забезпечивши їх зручним продуктом для швидкого створення модуля з найменшими затратами. Найбільш актуальним це рішення є для новостворених компаній, в яких є потреба в легкому, швидкому, централізованому зворотному зв'язку.

Метою даної випускної кваліфікаційної роботи є реалізація швидкої можливості комунікації між компанією та споживачем. При цьому важливо автоматизувати та систематизувати цей процес, щоб зробити його максимально ефективним та зручним, як для споживача так і для організації.

Розроблений продукт призначений для компаній, які потребують повноцінний та швидкий спосіб комунікації з клієнтами не витрачаючи для цього багато зусиль.

Для досягнення поставленої мети були визначені наступні задачі:

					IT61.140БАК.004 ПЗ	Арк.
						11
Змн.	Арк.	№ докум.	Підпис	Дата		

- аналіз існуючих модулів зворотного зв'язку;
- аналіз реакцій споживачів в цих модулях;
- об'єднання та систематизація інформації про різні модулі зворотного зв'язку для створення універсального;
- вибір найбільш ефективного програмного забезпечення для створення системи;
- розробка програмного забезпечення.

Таким чином, завданням даної випускної кваліфікаційної роботи є створення автоматизованої системи зворотного зв'язку для впровадження комунікації між клієнтами та компанією за допомогою створення свого власного блока оберненого зв'язку .

					IT61.140БАК.004 ПЗ	Арк.
						12
Змн.	Арк.	№ докум.	Підпис	Дата		

1 ОГЛЯД ІСНУЮЧИХ РІШЕНЬ

1.1 Існуючі модулі зворотного зв'язку

В даний момент розвитку інтернет індустрії обернений зв'язок є дуже важливою частиною компанії. В кожній організації він відрізняється своїм охопленням. Від невеликого блока з телефонами гарячих ліній, до спеціально створених форм для відгуків. Існують спеціальні сайти, які призначенні для централізованих відгуків. Розглянемо системи які спрямовані на збір загальної інформації про різні компанії з можливістю залишати відгуки.

Yelp – сайт для перегляду компаній, які займаються певним типом послуг, таких як аптеки або магазини з можливістю залишати та переглядати відгуки. [2]

Основні переваги Yelp:

- велика база компаній;
- можливість оцінити компанію.

Основні недоліки:

- складність в отриманні відповіді на запитання від компанії;
- не має основної інформації про компанію;
- компанія не може налаштувати сторінку з їх компанією;
- компанії можуть не знати про те, що вони є на сайті.

otzovik.com – web-сайт, де автори пишуть свої враження про все на світі, ставлять оцінки, дають корисні поради та ідеї, як вигідно і з користю витратити свій час і гроші, а також попереджають про можливі недоліки та проблеми, з якими можете зіткнутися при покупці або поїздки. [3]

Основні переваги otzovik.com:

- велика база компаній;
- можливість оцінити різні аспекти компанії.

Основні недоліки:

- відсутність інформації про зворотний зв'язок з компанією;
- складність в отриманні відповіді на запитання від компанії.

					IT61.140БАК.004 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		13

DOU – web-портал, де можна знайти новини, аналітичні статті та свіжу інформацію, пов'язану з інформаційними технологіями. На цьому сайті є окремий блок «Робота», де ти можеш знайти компанію в якій ти зацікавлений, переглянути відгуки про неї чи задати запитання. [4]

Основні переваги DOU:

- актуальна інформація про компанію;
- можливість оцінити різні аспекти компанії;
- компанії часто переглядають сайт, та відповідають на запитання.

Основні недоліки:

- вузька спеціалізація сайту;
- компанії які знаходяться на сайті більше цікавлять людей, які хочуть в них працювати, а ніж клієнтів.

Відгуки клієнтів - це лише частина зворотного зв'язку. Недоліком таких сайтів є складність самій компанії обробити відгуки, отримати конкретні запитання та надати відповідь на запитання. Також до існуючих рішень можна віднести різні часткові модулі оберненого зв'язку на спеціальних сайтах, які допомагають підтримувати зв'язок з клієнтом, але не є повноцінними, тобто спрямовані лише для отримання відгуку.

SiteGround – одна з компаній яка дає легко залишали відгуки про свій бренд і послуги. Як тільки ви оціните агента, натиснувши на будь-яку з п'яти зірочок і натиснувши кнопку «Оцінити», ви будете перенаправлені в інше вікно, де вас проситимуть залишити відгук. [5]

Основні переваги SiteGround:

- компанія отримує найважливіші дані про себе.

Основні недоліки:

- клієнти не мають змогу отримати відповідь на запитання;
- клієнти не можуть переглянути інші відгуки, щоб зробити висновки.

Huatt – додає спеціальну кнопку «зворотній зв'язок» для форм зворотного зв'язку. Після того, як ви натиснете цю кнопку, ви побачите поле, де ви можете залишити відгук про ваш досвід роботи з брендом. Таким чином, Huatt збирає

					ІТ61.140БАК.004 ПЗ	Арк.
						14
Змн.	Арк.	№ докум.	Підпис	Дата		

відгуки з декількох джерел, і це відіграє важливу роль в тому, що вони можуть дізнатися, хто їхні клієнти і чого вони хочуть. [6]

Основні переваги Hyatt:

- компанія отримує найважливіші дані про себе.

Основні недоліки:

- клієнти не мають змогу отримати відповідь на запитання;
- клієнти не можуть переглянути інші відгуки, щоб зробити висновки.

Google maps – має блок зворотного зв'язку, де клієнт вибравши на карті місце де він побував може оцінити його та залишити відгук. Користувач на картах зазвичай шукає не конкретну компанію, а заклади які знаходяться поблизу місця знаходження користувача.[7]

Основні переваги Google maps:

- зручний спосіб користувачу знайти інформацію про заклади поблизу;
- здатність оцінити заклад;
- компанія може зареєструвати інформацію про свій заклад.

Основні недоліки:

- компанії зазвичай не відповідають на відгуки;
- користувачі шукають не конкретну компанію, а заклади які знаходяться поблизу;
- блок не є повноцінним.

1.2 Обґрунтування доцільності розробки

Є два варіанти вибору отримання відповіді на питання, що саме хочуть клієнти від вашого продукту. Або намагатися вгадати, або звернутися до них, щоб зрозуміти що їм потрібно. Виявляється, що багато компаній сьогодні ще все-таки намагаються вгадати. Дослідження показали, що 42% компаній не враховують і не збирають відгуки своїх клієнтів.[8]

Причиною такої статистики є навіть не те що компанії ігнорують думку клієнтів, а те що не має можливості або не є доцільним створювати окремий сайт з

					ІТ61.140БАК.004 ПЗ	Арк.
						15
Змн.	Арк.	№ докум.	Підпис	Дата		

модулем зворотного зв'язку. Тобто є організації яким не потрібен веб сайт , але було б дуже важливо отримати думку клієнтів, або невеликі компанії яким занадто дорого створювати свій веб сайт, а в просторі Інтернет повноцінних систем які були б сфокусовані на зворотний зв'язок виявлено не було.

З іншого боку навіть ті компанії, у яких наявні елементи оберненого зв'язку, мають їх не в повному розмірі. Тобто, в деяких з них, для обробки залишених відгуків не існує інструмента надання відповіді онлайн чи за допомогою електронної пошти. Іншою проблемою є відсутність розділу поширенні питання, яка може втричі зменшити кількість запитань до компанії та допомогти багатьом клієнтам. Причиною створення таких неповноцінних модулів є збереження мінімалістичного інтерфейсу, для того щоб отримати лише найважливішу інформацію та зберегти веб сайт простим для використання. Саме тому розроблений програмний продукт дозволить створити швидкий обернений зв'язок та буде спеціалізована на ньому. Нові компанії зможуть безкоштовно реєструватися в системі, та створювати свій модуль зв'язку. Великі компанії зможуть зберігати мінімалізм свого сайту, а також додатково отримати повноцінний потужний модуль для автоматизації процесу зворотного зв'язку. В той же час клієнти отримають систематизований продукт де можливо переглянути компанії які їх цікавлять.

Висновки до розділу

В цьому розділі було детально проаналізовано існуючі модулі зворотного зв'язку. В результаті не було виявлено системи, яка могла би створити повноцінний процес комунікації між компанією та клієнтом. Було визначені сильні та слабкі сторони окремих модулів, а також визначено доцільність створення нового продукту.

					IT61.140БАК.004 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		16

2 АНАЛІЗ ОСОБЛИВОСТЕЙ СИСТЕМИ ЗВОРОТНОГО ЗВ'ЯЗКУ

2.1 Опис та аналіз предметного середовища

Автоматизована система зворотного зв'язку являє собою web-сайт цілю якого є створення можливості зв'язку між клієнтом та компанією, а також автоматизацією цього процесу.

Найпопулярнішою та найефективнішою архітектурою для цієї системи є клієнт-сервер. Вона складається з таких компонентів:

- сервер, який отримує та віддає дані з клієнтської сторони та оброблює їх в даній системі сервер буде також відповідати за відправлення нотифікацій користувачам відповідно до різних дій;
- клієнт, який отримує дані з сервера, оброблює та відображає їх через зручний інтерфейс - інтерфейс повинен підтримуватись у всіх браузерях, бути адаптивним, семантичним, компонентним;
- база даних, яка зберігає дані - повинна містити індекси, які дозволяють пришвидшити процес пошуку відповідних даних. Також повинна мати завдання, які через деякі періоди часу повинні оброблювати дані, щоб зберігати їх актуальність.

Систему зворотного зв'язку потрібно розглянути відносно двох ролей, власника компанії та відносно клієнта компанії.

Основною перевагою для власника компанії є швидкість, яка дозволяє створити цілий модуль зворотного зв'язку лише за декілька хвилин. Панель адміністрування компанії містить необхідний функціонал для легкого наповнення блоку зворотного зв'язку, а ролева модель допоміжних модераторів компанії, дозволить назначити управління зворотним зв'язком компанії вашому довіреному представникові.

Переваги для клієнта полягають у зручному інтерфейсі системи з можливістю перегляду статистики та рейтингу компаній. А також надасть можливість швидкого зв'язку щ компанією для вирішення проблем клієнта.

					ІТ61.140БАК.004 ПЗ	Арк.
						17
Змн.	Арк.	№ докум.	Підпис	Дата		

2.2 Функціональні вимоги до системи

Система зворотного зв'язку з клієнтом складається з різних блоків зворотного зв'язку, які були об'єднані шляхом аналізу та систематизації найбільш важливих, існуючих блоків в різних організаціях.

Основні ролі в системі :

- неавторизований користувач – учасник системи, який не пройшов процедуру автентифікації;
- авторизований користувач – учасник системи, який пройшов процедуру автентифікації;
- адміністратор – учасник системи, який створив компанію;
- модератор – учасник системи, призначений адміністратором, який перевіряє коректність повідомлень і приймає рішення на відправку відповіді, може назначити відповідального на тему і відправляти йому запитання в яких відповідальний є спеціалістом;
- відповідальний – учасник системи, який відповідає на запитання призначені йому модератором.

Продукт повинен складатися з модулів, які дозволять адмініструвати та автоматизувати процес.

Адміністративний модуль є основна частина системи, що надає можливість створювати інформацію про компанію, заповнювати гарячі лінії, налаштовувати найчастіші запитання, а також назначати модераторів та відповідальних, яким надано доступ до управління зворотним зв'язком компанії, наприклад можливість відповідати на запитання чи позначати їх як найчастіші.

Другим важливим модулем є статистика та рейтинг компанії, яка повинна вираховувати кількість наданих запитань та час за який ці запитання були відповіді, оцінювати кількість відгуків та активність користувачів.

Третім важливим модулем є блок нотифікації. На різні дії, такі які відправлення повідомлення, прострочення часу на відповідь на запитання,

					IT61.140БАК.004 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		18

призначення модератора та відповідального, блокування користувача повинно приходити повідомлення на пошту.

Також система містить статичні сторінки, на яких відображена основна інформація, та є можливість взаємодіяти з компанією, наприклад залишити відгук чи задати запитання.

Основна мета розробки цього продукту для власника компанії є швидкість в створенні повноцінного, автоматизованого модулю зворотного зв'язку без значних затрат.

Основна мета створенні цього продукту для клієнта компанії є у систематизації зворотного зв'язку в одному місці зі зручним інтерфейсом, а також з можливістю перегляду статистики та рейтингу компаній

Процес зворотного зв'язку представляє собою взаємодію користувачами системи і спеціалістами в різних напрямленнях, які визначені в системі.

Функціональними вимогами є створення таких блоків системи:

- блок автентифікації :
 - а) реєстрація;
 - б) можливість використання сторонніх сервісів авторизації;
 - в) авторизація;
- неавторизований користувач може зареєструватися в системі та пройти процедуру автентифікації для того, щоб зайти в свій унікальний акаунт;
- блок управління компанією :
 - а) корисна інформація;
 - б) гарячі лінії;
 - в) теми для запитань;
- адміністратор компанії може заповнювати корисну інформацію про компанію та гарячі лінії;
- адміністратор та модератор компанії може задавати теми для запитань до компанії;
- блок запитання :
 - а) перегляд запитань;

- б) відповідь на запитання;
- в) фільтрація запитань;
- г) найчастіші запитання;
- г) відправка нагадування на пошту, якщо на запитання не дати відповідь впродовж зазначеного адміністратором критичного часу;
- д) управління запитаннями;
- е) відправка повідомлення користувачу на пошту, коли на його запитання надали відповідь;
- є) відправка відповідальному на пошту, коли йому назначили запитання;
 - користувач може відправити запитання та переглянути його;
 - адміністратор, модератор, відповідальний може самостійно заповнити форму відповіді на запитання і відправити ініціатору (користувачеві) запитання;
 - адміністратор, модератор, відповідальний може прийняти рішення, що запитання некоректно і видалити запитання без відповіді ініціатору (користувачеві) за запитанням;
 - адміністратор, модератор, відповідальний може опублікувати запитання з відповіддю на формі «Найчастіші запитання», як нове запитання в процесі модерації, так і раніше відправлене ініціатору запитання, вибравши його зі списку;
 - адміністратор, модератор, відповідальний може переглянути запитання, за якими пройшов термін надання відповіді на запитання, і відправлені відповідальним на модератора без заповнення відповіді на запитання;
 - адміністратор, модератор може призначити нового відповідального для запитання, і відправити на обробку для нового відповідального. Система після призначення нового відповідального за запитанням, автоматично відправляє новому відповідальному лист на електронну пошту про те що на нього призначено запитання;
 - відповідальний може переглянути запитання, написати відповідь на запитання, відправити запитання модератору та адміністратору без відповіді на запитання. Система автоматично відстежує кількість днів з моменту створення

					IT61.140БАК.004 ПЗ	Арк.
						20
Змн.	Арк.	№ докум.	Підпис	Дата		

запитання і, якщо на запитання не відправлена відповідь, відправляє лист попередження відповідальному, що залишився один робочий день для відповіді на запитання;

- якщо протягом заданого адміністратором (критичних днів на відповідь) робочих днів відповідь на запитання не створена, система переводить запитання в статус «Построчено» і відправляє лист на електронну пошту модератора і відповідального, що відповідь на запитання построчено. Відповідальний фахівець заповнює відповідь на запитання користувача і відправляє для перевірки модератору;

- адміністратор та модератор може перевірити коректність питання і відповіді на запитання, відкоригувати запитання, прийняти рішення про неактуальність запитання, видалити відповідь, прийняти рішення про відправку запитання ініціатору, якщо модератор прийняв рішення відправити запитання модератору, система автоматично відправляє на електронну пошту ініціатору запитання повідомлення, що на ваше запитання відправлено відповідь;

- блок відгуку:

- а) перегляд відгуків;
- б) відповіді на відгуки;
- в) оцінка відгуків;
- г) відповідь на відгук;
- г) редагувати відгук.

- користувач, модератор, адміністратор, відповідальний можуть створювати відгуки, оцінювати їх та відповідати на них.

Діаграма процесу зворотного зв'язку в блоці «запитання» наведено у кресленику ІТ61.140БАК.004 Д2.

2.3 Нефункціональні вимоги до системи

Нефункціональними вимогами до системи є :

					ІТ61.140БАК.004 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		21

– повторне використання компонентів – використання для нових розроблень фрагментів коду, здобутих у процесі розроблення інших блоків системи. Це дозволить пришвидшити процес тестування. Протестований компонент може використовуватись по всьому проекту без його повторного тестування. Якщо ви виявите проблеми, у вас є менше місць для пошуку причини, оскільки, швидше за все, проблема полягає в новій створеній вами частині;

– кросбраузерність – система повинна підтримуватися у всіх найпопулярніших браузерах для того, щоб будь-який клієнт зміг її використовувати повноцінно;

– масштабованість – система повинна мати змогу безпроблемного збільшення потоку споживачів. Система має бути орієнтована на участь як великих так і не великих компаній і витримати прилив їх клієнтів;

– семантичність – система повинна мати змогу бути знайденою на перших сторінках в пошукових системах;

– адаптивність – система повинна бути доступною як на комп'ютерах так і на мобільних телефонах;

– кросплатформність – здатність системи бути розміщеною на будь якій операційній системі також буде плюсом.

Висновки до розділу

В даному розділі було проаналізовано особливості системи зворотного зв'язку та було висунуті функціональні та нефункціональні вимоги системи. Це дозволить створити гнучку архітектуру і попередить виникнення критичних проблем .

					IT61.140БАК.004 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		22

3 СЦЕНАРІЇ ВИКОРИСТАННЯ СИСТЕМИ

3.1 Сценарії використання системи

На креслениках ІТ61.140БАК.004 Д1.1 - ІТ61.140БАК.004 Д1.2 наведені діаграми варіантів використання (діаграми прецедентів).

Таблиця 3.1 – Варіанти використання системи

Варіант використання	Назва використання
Case-1	Перегляд меню авторизації
Case-1.1	Авторизація
Case-1.2	Реєстрація
Case-2	Перегляд моїх та збережених компаній
Case-3	Перегляд інформації про компанію
Case-4	Перегляд відгуків
Case-4.1	Створити відгук
Case-4.2	Редагувати відгук
Case-4.3	Видалити відгук
Case-4.4	Відповісти на відгук
Case-4.5	Оцінити відгук
Case-5	Перегляд статистики та рейтингу компаній
Case-6	Додати компанію в збережені
Case-7	Перегляд моїх запитань
Case-7.1	Поставити запитання
Case-8	Створити компанію
Case-8.1	Редагувати компанію
Case-8.2	Видалити компанію
Case-9	Перегляд модераторів компанії
Case-9.1	Додати модератора до компанії

Варіант використання	Назва використання
Case-9.2	Видалити модератора
Case-10	Перегляд тем компанії
Case-10.1	Додати тему до компанії
Case-10.2	Редагувати тему
Case-10.3	Видалити тему
Case-11	Перегляд відповідальних компанії
Case-11.1	Додати відповідального на тему
Case-11.2	Видалити відповідального
Case-12	Перегляд всіх запитань
Case-12.1	Перегляд прострочених запитань
Case-12.2	Надання відповіді на запитання
Case-12.3	Призначити відповідального за запитання
Case-12.4	Закріпити запитання в розділі найчастіші запитання
Case-13	Задання критичного часу на відповідь
Case-14	Повернути питання модератору

У таблицях 3.2-3.36 приведені описи кожного з варіантів використання.

Таблиця 3.2 – Варіант використання Case-1

Назва	Перегляд меню авторизації
Опис	Будь-який неавторизований користувач має право переглянути меню авторизації.
Учасники	Неавторизований користувач.
Передумови	
Постумови	Відображення сторінки авторизації.
Основний сценарій	1) Користувач натиснув посилання для переходу на будь-яку сторінку web-сайта. 2) Відображається сторінка авторизації.

Розширення	
------------	--

Таблиця 3.3 – Варіант використання Case-1.1

Назва	Авторизація.
Опис	Будь-який зареєстрований користувач має право авторизуватися в систему.
Учасники	Неавторизований користувач.
Передумови	Користувач перейшов на сторінку авторизації.
Постумови	Система авторизує користувача.
Основний сценарій	1) Користувач знаходиться на сторінці для авторизації. 2) Користувач вводить свою пошту та пароль. 3) Користувач натискає кнопку “Вхід”. 4) Система авторизує користувача.
Розширення	1) Система проводить операцію валідації та повідомляє, що введені дані некоректні. 2) Відображення сторінки авторизації з повідомленням щодо некоректно введених даних.

Таблиця 3.4 – Варіант використання Case-1.2

Назва	Реєстрація
Опис	Будь-який користувач має право зареєструватися в системі.
Учасники	Неавторизований користувач.
Передумови	Користувач натиснув на кнопку «Реєстрація»
Постумови	Реєстрація користувача.
Основний сценарій	1) Користувач заповнює поля обов’язкові для реєстрації. 2) Користувач натискає кнопку “Зареєструватися”. 3) Система реєструє користувача.

Розширення	<p>1) Система проводить операцію валідації та повідомляє, що введені дані некоректні.</p> <p>2) Відображення сторінки реєстрації з повідомленням щодо некоректно введених даних.</p>
------------	--

Таблиця 3.5 – Варіант використання Case-2

Назва	Перегляд моїх та збережених компаній
Опис	Будь-який авторизований користувач може переглянути компанії, які він створив чи додав в збережені.
Учасники	Авторизований користувач.
Передумови	
Постумови	Демонстрація списку компаній.
Основний сценарій	<p>1) Користувач переходить на сторінку «Мої компанії».</p> <p>2) Відображено список компаній.</p>
Розширення	

Таблиця 3.6 – Варіант використання Case-3

Назва	Перегляд інформації про компанію
Опис	Будь-який користувач має право переглянути інформацію про компанію.
Учасники	Будь-який користувач.
Передумови	
Постумови	Демонстрація інформації про компанію.
Основний сценарій	<p>1) Користувач переходить за посиланням на компанію.</p> <p>2) Відображено інформацію про компанію.</p>
Розширення	

Таблиця 3.7 – Варіант використання Case-4

Назва	Перегляд відгуків
Опис	Будь-який користувач має право переглянути відгуки про компанію.
Учасники	Будь-який користувач.
Передумови	
Постумови	Демонстрація відгуків про компанію.
Основний сценарій	1) Користувач переходить за посиланням на компанію. 2) Відображено відгуки про компанію.
Розширення	

Таблиця 3.8 – Варіант використання Case-4.1

Назва	Створити відгук
Опис	Будь-який авторизований користувач має право створити відгук про компанію.
Учасники	Авторизований користувач, адміністратор, модератор, відповідальний.
Передумови	Перейти на сторінку перегляду інформації про компанію.
Постумови	Відгук був створений.
Основний сценарій	1) Перейти на сторінку перегляду інформації про компанію. 2) Натиснути кнопку «Надати відгук». 3) Ввести текст відгуку. 4) Натиснути кнопку «Відправити». 5) Система добавила та відобразила відгук.
Розширення	1) Система записує відгук до бази даних. 2) Система створює зв'язок між відгуком та компанією.

Таблиця 3.9 – Варіант використання Case-4.2

Назва	Редагувати відгук
Опис	Користувач може відредагувати свій відгук.
Учасники	Авторизований користувач, адміністратор, модератор, відповідальний.
Передумови	1) Бути власником відгуку. 2) Перейти на сторінку перегляду відгуків.
Постумови	Система редагує відгук в базі даних.
Основний сценарій	1) Перейти на сторінку інформації про компанії. 2) Натиснути кнопку «Редагувати» поряд з відгуком який залишив сам користувач. 3) Змінити текст відгуку. 4) Натиснути кнопку «Підтвердити». 5) Система оновила та відобразила відгук.
Розширення	1) Система оновила текст відгуку в базі даних. 2) Система відобразила признак відредактованості біля відгуку.

Таблиця 3.10 – Варіант використання Case-4.3

Назва	Видалити відгук
Опис	Користувач який створив відгук може його видалити
Учасники	Адміністратор, модератор, відповідальний, авторизований користувач який створив цей відгук.
Передумови	Користувач створив відгук. Перейти на сторінку перегляду відгуків.
Постумови	Система видаляє відгук з бази даних
Основний сценарій	1) Перейти на сторінку інформації про компанії.

	<p>2) Натиснути кнопку «Видалити» поряд з відгуком який залишив сам користувач.</p> <p>3) Натиснути кнопку «Підтвердити».</p> <p>4) Система видалила відгук та відобразила список відгуків без нього.</p>
Розширення	Система видаляє всі відповіді на відгук, який був видалений.

Таблиця 3.11 – Варіант використання Case-4.4

Назва	Відповісти відгук
Опис	Будь-який авторизований користувач може залишити відповідь на відгук.
Учасники	Адміністратор, модератор, відповідальний, авторизований користувач.
Передумови	<p>1) Користувач авторизований.</p> <p>2) Користувач перейшов на сторінку перегляду інформації про компанію.</p> <p>3) Список відгуків має відгук.</p>
Постумови	Система додала та відобразила відповідь на відгук.
Основний сценарій	<p>1) Перейти на сторінку перегляду інформації про компанію.</p> <p>2) Натиснути кнопку «Відповісти».</p> <p>3) Ввести текст відповіді.</p> <p>4) Натиснути кнопку «Відправити».</p> <p>5) Система добавила та відобразила відповідь на відгук.</p>
Розширення	<p>1) Система записує відповідь на відгук до бази даних.</p> <p>2) Система додає зв'язок між відгуком та відповіддю на відгук.</p>

Таблиця 3.12 – Варіант використання Case-4.5

Назва	Оцінити відгук
Опис	Будь-який авторизований користувач може оцінити відгук.
Учасники	Адміністратор, модератор, відповідальний, авторизований користувач.
Передумови	1) Користувач авторизований. 2) Користувач перейшов на сторінку перегляду інформації про компанію. 3) Список відгуків має відгук.
Постумови	Система записала та відобразила оцінку на відгук.
Основний сценарій	1) Перейти на сторінку перегляду інформації про компанію. 2) Натиснути кнопку «Оцінити». 3) Система добавила та відобразила оцінку на відгук.
Розширення	1) Система записує оцінку до бази даних. 2) Система створює зв'язок між оцінкою та відгуком.

Таблиця 3.13 – Варіант використання Case-5

Назва	Перегляд статистики та рейтингу компаній.
Опис	Будь-який користувач може переглянути статистику та рейтинг компанії.
Учасники	Будь-який користувач.
Передумови	
Постумови	Демонстрація статистики компанії.
Основний сценарій	1) Перейти на сторінку інформації про компанію. 2) Система відображає статистику та рейтинг по компанії.
Розширення	

Таблиця 3.14 – Варіант використання Case-6

Назва	Додати компанію в збережені
Опис	Будь-який авторизований користувач може додати компанію в збережені.
Учасники	Авторизований користувач, адміністратор, модератор, відповідальний.
Передумови	1) Користувач перейшов на сторінку перегляду інформації про компанію 2) Користувач не є адміністратором цієї компанії.
Постумови	Система додала компанію до збережених користувача.
Основний сценарій	1)Перейти на сторінку перегляду інформації компанії. 2. Натиснути функціональний елемент «Додати компанію в збережені».
Розширення	Система створила зв'язок між компанією та користувачем.

Таблиця 3.15 – Варіант використання Case-7

Назва	Перегляд моїх запитань
Опис	Будь-який авторизований користувач може переглянути запитання які він задав до компанії.
Учасники	Авторизований користувач, адміністратор, модератор, відповідальний.
Передумови	1) Користувач авторизований в системі. 2) Користувач перейшов на сторінку інформації про компанію.
Постумови	Демонстрація списку запитань користувача.
Основний сценарій	1) Перейти на сторінку інформації користувача. 2) Натиснути кнопку «Мої запитання». 3) Система відображає список запитань.

Розширення	
------------	--

Таблиця 3.16 – Варіант використання Case-7.1

Назва	Задати запитання
Опис	Будь-який авторизований користувач може задати запитання до компанії за темою, яку він обрав.
Учасники	Авторизований користувач, адміністратор, модератор, відповідальний.
Передумови	1) Користувач авторизований в системі. 2) Користувач перейшов на сторінку перегляду його запитань.
Постумови	Користувач відправив запитання.
Основний сценарій	1) Перейти на сторінку перегляду запитань. 2) Натиснути кнопку «Задати запитання». 3) Вибрати тему із списку запропонованих. 4) Вести текст запитання. 5) Натиснути кнопку «Відправити запитання».
Розширення	1) Система додає запитання до бази даних. 2) Система створює зв'язок між запитанням та компанією.

Таблиця 3.17 – Варіант використання Case-8

Назва	Створити компанію
Опис	Будь-який авторизований користувач може створити компанію.
Учасники	Авторизований користувач.
Передумови	Користувач зареєстрований в системі.

	Користувач знаходиться на сторінці перегляду всіх компаній.
Постумови	Система створює запис про компанію в базі даних.
Основний сценарій	1) Система відображає сторінку створення компанії. 2) Користувач вводить інформацію. 3) Користувач натискає кнопку «Створити компанію». 4) Система створює запис про компанію в базі даних. 5) Відображається сторінка з інформацією про компанію.
Розширення	Система призначає користувача адміністратором компанії.

Таблиця 3.18 – Варіант використання Case-8.1

Назва	Редагувати компанію.
Опис	Адміністратор компанії може редагувати інформацію.
Учасники	Адміністратор.
Передумови	Користувач є адміністратором компанії.
Постумови	Збереження нової інформації про компанію.
Основний сценарій	1) Адміністратор натиснув кнопку редагування на сторінці перегляду інформації про компанію. 2) Адміністратор вводить нову інформацію. 3) Адміністратор натискає кнопку «Зберегти зміни». 4) Система замінює запис про компанію в базі даних. 5) Відображається сторінка з інформацією про компанію.
Розширення	

Таблиця 3.19 – Варіант використання Case-8.2

Назва	Видалити компанію.
Опис	Адміністратор компанії може видалити компанію.
Учасники	Адміністратор.
Передумови	Користувач є адміністратором компанії. Користувач знаходиться на сторінці перегляду компанії або перегляду всіх його компаній.
Постумови	Видалити запис про компанію з бази даних.
Основний сценарій	1) Адміністратор натиснув кнопку видалення на сторінці перегляду інформації про компанію. 2) Адміністратор натискає кнопку «Підтвердити» в діалоговому вікні. 3) Відображається сторінка з усіма компаніями.
Розширення	

Таблиця 3.20 – Варіант використання Case-9

Назва	Перегляд усіх модераторів компанії
Опис	Адміністратор компанії може переглядати список модераторів компанії.
Учасники	Адміністратор.
Передумови	Користувач є адміністратором компанії.
Постумови	Демонстрація списку усіх модераторів
Основний сценарій	1) Адміністратор переходить за посиланням на сторінку всіх модераторів. 2) Відображено список модераторів.
Розширення	

Таблиця 3.21 – Варіант використання Case-9.1

Назва	Додати модератора до компанії
Опис	Адміністратор компанії може призначити модератора до компанії.
Учасники	Адміністратор.
Передумови	Користувач є адміністратором компанії. Користувач знаходиться на сторінці перегляду інформації про компанію.
Постумови	Призначення користувача модератором компанії.
Основний сценарій	1) Перейти на сторінку модераторів компанії. 2) Натиснути кнопку «Додати модератора». 3) Вибрати користувача серед зареєстрованих користувачів. 4) Натиснути кнопку «Підтвердити». 5) Система призначає модератора компанії.
Розширення	Система створює користувачу признак модератора компанії

Таблиця 3.22 – Варіант використання Case-9.2

Назва	Видалити модератор
Опис	Адміністратор компанії може видалити модератора з компанії.
Учасники	Адміністратор.
Передумови	Користувач є адміністратором компанії.
Постумови	Користувач втратив признак модератора в компанії
Основний сценарій	1) Перейти на сторінку модераторів компанії. 2) Натиснути кнопку «Видалити модератора» біля ФІО модератора якого потрібно видалити. 3) Натиснути кнопку «Підтвердити».

	4) Система видаляє модератора з компанії.
Розширення	Система знімає признак модератора компанії з користувача.

Таблиця 3.23 – Варіант використання Case-10

Назва	Переглянути основні теми до блоку запитань.
Опис	Адміністратор та модератор компанії можуть переглянути теми на які можна поставити запитання
Учасники	Адміністратор, модератор.
Передумови	Користувач є адміністратором або модератором компанії. Користувач знаходиться на сторінці перегляду інформації про компанію.
Постумови	Демонстрація тем до блоку запитань
Основний сценарій	1) Адміністратор переходить за посиланням на сторінку всіх тем. 2) Відображено список тем.
Розширення	

Таблиця 3.24 – Варіант використання Case-10.1

Назва	Додати тему до компанії.
Опис	Адміністратор та модератор компанії може створити тему для списку тем в розділі запитання.
Учасники	Адміністратор, модератор.
Передумови	Користувач є адміністратором або модератором компанії.
Постумови	Система записує тему до бази даних.
Основний сценарій	1) Перейти на сторінку тем компанії. 2) Натиснути кнопку «Додати».

	3) Ввести назву теми. 4) Натиснути кнопку «Підтвердити». 5) Система створює нову тему. 6) Система відображає список тем.
Розширення	1) Система записує тему до бази даних. 2) Система створює зв'язок між темою та компанією.

Таблиця 3.25 – Варіант використання Case-10.2

Назва	Редагувати тему запитань компанії
Опис	Адміністратор та модератор компанії може редагувати тему.
Учасники	Адміністратор, модератор.
Передумови	Користувач є адміністратором або модератором компанії.
Постумови	Тема була відредагована.
Основний сценарій	1) Перейти на сторінку тем компанії. 2) Натиснути кнопку «Додати». 3) Ввести назву теми. 4) Натиснути кнопку «Підтвердити». 5) Система відображає список тем.
Розширення	1) Система записує тему до бази даних. 2) Система створює зв'язок між темою та компанією.

Таблиця 3.26 – Варіант використання Case-10.3

Назва	Видалити тему запитань компанії
Опис	Адміністратор та модератор компанії може редагувати тему для списку тем в розділі запитання.
Учасники	Адміністратор, модератор.

Передумови	Користувач є адміністратором або модератором компанії.
Постумови	Користувач втратив признак модератора в компанії
Основний сценарій	1) Перейти на сторінку тем компанії. 2) Натиснути кнопку «Додати». 3) Ввести назву теми. 4) Натиснути кнопку «Підтвердити». 5) Система відображає список тем.
Розширення	1) Система записує тему до бази даних. 2) Система створює зв'язок між темою та компанією.

Таблиця 3.27 – Варіант використання Case-11

Назва	Перегляд усіх відповідальних компанії
Опис	Адміністратор, модератор компанії можуть переглядати список відповідальних компанії.
Учасники	Адміністратор, модератор.
Передумови	Користувач є адміністратором або модератором та знаходиться на сторінці інформації про компанії.
Постумови	Демонстрація списку усіх відповідальних
Основний сценарій	1) Адміністратор або модератор натискає кнопку «Відповідальні компанії». 2) Адміністратор або модератор переходить за посиланням на сторінку всіх відповідальних. 3) Відображено список відповідальних.
Розширення	

Таблиця 3.28 – Варіант використання Case-11.1

Назва	Назначити відповідального на тему
Опис	Адміністратор, модератор компанії може призначити відповідального на тему.
Учасники	Адміністратор, модератор.
Передумови	Користувач є адміністратором або модератором компанії та знаходиться на сторінці «Усі відповідальні».
Постумови	Призначення користувача відповідальним по темі.
Основний сценарій	1) Перейти на сторінку відповідальних компанії. 2) Натиснути кнопку «Додати відповідального». 3) Вибрати користувача серед зареєстрованих користувачів. 4) Вибрати тему на яку потрібно призначити відповідального. 4) Натиснути кнопку «Підтвердити». 5) Система призначає відповідального компанії.
Розширення	Система створює зв'язок між користувачем, якого обрали відповідальним, компанією та темою.

Таблиця 3.29 – Варіант використання Case-11.2

Назва	Видалити відповідального за тему
Опис	Адміністратор, модератор компанії може видалити відповідального з компанії.
Учасники	Адміністратор, модератор.
Передумови	Користувач є адміністратором або модератором компанії та знаходиться на сторінці «Усі відповідальні».

Постумови	Користувач втратив признак відповідального в компанії
Основний сценарій	1) Перейти на сторінку «Усі відповідальні» компанії. 2) Натиснути кнопку «Видалити модератора» біля ФІО модератора якого потрібно видалити. 3) Натиснути кнопку «Підтвердити». 4) Система видаляє відповідального за тему.
Розширення	Система видаляє признак відповідального за тему з користувача.

Таблиця 3.30 – Варіант використання Case-12

Назва	Перегляд всіх запитань
Опис	Адміністратор, модератор компанії можуть переглядати список всіх запитань до компанії.
Учасники	Адміністратор, модератор, відповідальний.
Передумови	Користувач є адміністратором, модератором або відповідальним та знаходиться на сторінці інформації про компанії.
Постумови	Демонстрація списку усіх запитань до компанії.
Основний сценарій	1) Адміністратор або модератор або відповідальний натискає кнопку «Всі запитання». 2) Адміністратор або модератор, або відповідальний переходить за посиланням на сторінку всіх запитань. 3) Відображено список всіх запитань.
Розширення	

Таблиця 3.31 – Варіант використання Case-12.1

Назва	Перегляд прострочених запитань
Опис	Адміністратор, модератор компанії можуть переглядати тільки прострочені запитання.
Учасники	Адміністратор, модератор, відповідальний.
Передумови	1) Користувач є адміністратором, модератором або відповідальним. 2) Користувач знаходиться на сторінці усіх запитань.
Постумови	Демонстрація списку усіх прострочених запитань до компанії.
Основний сценарій	1) Адміністратор або модератор або відповідальний натискає кнопку «Прострочені запитання». 2) Система фільтрує всі запитання та виводить тільки простроченні. 3) Відображено список всіх запитань.
Розширення	

Таблиця 3.32 – Варіант використання Case-12.2

Назва	Відповідь на запитання
Опис	Адміністратор, відповідальний, модератор можуть відповісти на запитання.
Учасники	Адміністратор, відповідальний, модератор.
Передумови	1) Користувач авторизований в системі. 2) Користувач є відповідальним або модератором, або адміністратором цієї компанії. 3) Користувач перейшов на сторінку перегляду його запитань.
Постумови	Надано відповідь на запитання.

Основний сценарій	1) Перейти на сторінку перегляду запитань. 2) Натиснути на запитання зі списку всіх запитань, на яке потрібно надати відповідь. 3) Вести текст відповідь. 4) Натиснути кнопку «Відправити».
Розширення	1) Система додає відповідь до бази даних. 2) Система створює зв'язок між запитанням та відповіддю.

Таблиця 3.33 – Варіант використання Case-12.3

Назва	Закріпити запитання в розділі «Найчастіші запитання»
Опис	Адміністратор, відповідальний, модератор можуть додати запитання в розділ «Найчастіші запитання».
Учасники	Авторизований користувач, адміністратор, модератор, відповідальний.
Передумови	1) Користувач є відповідальним або модератором, або адміністратором цієї компанії. 2) Користувач переглядає запитання на яке було надано відповідь.
Постумови	Система додає запитання в розділ «Найчастіші запитання»
Основний сценарій	1) Перейти на сторінку перегляду запитань. 2) Натиснути на запитання яке знаходиться в статусі «Надано відповідь». 3) Натиснути на кнопку «Додати в найчастіші». 4) Система додала запитання в «Найчастіші запитання».
Розширення	Система додає признак «найчастіше» до запитання в базі даних.

Таблиця 3.34 – Варіант використання Case-13

Назва	Призначити відповідального на запитання.
Опис	Адміністратор та модератор можуть відправити запитання відповідальному.
Учасники	Адміністратор, модератор.
Передумови	1) Користувач авторизований в системі. 2) Користувач є модератором, або адміністратором цієї компанії. 3) Користувач переглядає запитання на яке не було надано відповіді.
Постумови	Система назначає відповідального на запитання.
Основний сценарій	1) Перейти на сторінку перегляду запитань. 2) Натиснути на запитання яке знаходиться в статусі «Нове». 3) Натиснути на галочку «Відправити відповідальному». 4) Система призначила та відобразила запитання відповідальному.
Розширення	1) Система відправляє нотифікацію відповідальному про те, що йому назначене нове запитання. 2) Система створює зв'язок між запитанням та відповідальним.

Таблиця 3.35 – Варіант використання Case-14

Назва	Задання критичного часу на відповідь
Опис	Адміністратор може задати час за який на запитання повинно бути надана відповідь або запитання буде прострочене.
Учасники	Адміністратор.

Передумови	1) Користувач авторизований в системі. 2) Користувач є адміністратором компанії. 3) Користувач знаходиться на сторінці інформації про компанію.
Постумови	Система назначає кількість днів на запитання для компанії.
Основний сценарій	1) Перейти на сторінку інформації про компанію. 2) Натиснути на функціональний елемент «час на надання відповіді». 3) Ввести кількість днів. 4) Натиснути «Зберегти». 5) Система призначила кількість днів на відповідь.
Розширення	1) Система добавляє час на відповідь до компанії в базі даних. 2) Система створює зв'язок між запитанням та компанією.

Таблиця 3.36 – Варіант використання Case-15

Назва	Повернути запитання модератору
Опис	Відповідальний може відправити запитання модератору.
Учасники	Відповідальний.
Передумови	1) Користувач авторизований в системі. 2) Користувач переглядає запитання на яке не було надано відповіді. 3) Користувач є відповідальним за запитання.
Постумови	Система відображує функціональний елемент «увага» для модератора системи біля поверненого запитання.

Основний сценарій	1) Перейти на сторінку перегляду запитань. 2) Користувач є відповідальним за запитання. 3) Користувач натискає галочку, повернути запитання. 4) Користувач вводить причину повернення запитання. 5) Користувач натискає кнопку «Відправити». 6) Система додає признак «повернено» до повідомлення та зображує функціональний елемент «увага» біля запитання.
Розширення	1) Система відправляє нотифікацію модератору про те, що йому повернено запитання. 2) Система додає признак «повернено модератору» до повідомлення.

Висновки до розділу

В цьому розділі було описано основні сценарії в яких взаємодіють зовнішні сутності або актори. Було визначено загальні межі і контекст модельованої предметної області на початкових етапах проектування системи, сформульовано загальні вимоги до функціонального поведінки проектованої системи, розроблено вихідну концептуальну модель системи для її подальшої деталізації у формі логічних і фізичних моделей.

4 РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

4.1 Вибір технологій розробки

Вибираючи технологічний стек для цього проекту, важливо підібрати правильні інструменти, які прискорять розвиток та зменшать витрати. Посилаючись на нефункціональні вимоги, а найважливішими є повторне використання компонентів та розширюваність, я вирішив обрати клієнт – серверну архітектуру. Де сервером буде слугувати фреймворк ASP .Net core, а клієнтською частиною – Angular.

Angular - це платформа для веб-розробок, створена в TypeScript, яка надає розробникам надійні інструменти для створення клієнтської сторони веб-додатків. Основною причиною вибору Angular є те, що цей інструмент інтерфейсу оснащений надійними компонентами, які допомагають розробникам писати читабельний, тестований та простий у користуванні код. [9]

До сильних сторін Angular відносяться:

- чудова екосистема сторонніх компонентів. Популярність Angular призвела до появи тисяч додаткових інструментів та компонентів, які можна використовувати в програмах Angular. В результаті можна отримати додаткові функціональні можливості та підвищення продуктивності;
- компонентна архітектура. У другій версії Angular перейшов від MVC до компонентної архітектури. Відповідно до цієї архітектури додаток розділений на незалежні логічні та функціональні компоненти. Ці компоненти можна легко замінити та роз'єднати, а також повторно використати в інших частинах програми. Крім того, незалежність компонентів полегшує тестування веб-програми та забезпечує безперебійність роботи кожного компонента.

ASP.NET Core - це крос-платформна платформа з відкритим кодом для створення сучасних хмарних веб-додатків для Windows, macOS або Linux.[10]

ASP.NET Core надає такі переваги:

- легкість в створенні веб-інтерфейсу та веб-API;
- підтримка кросплатформності;

					IT61.140БАК.004 ПЗ	Арк.
						46
Змн.	Арк.	№ докум.	Підпис	Дата		

- підтримка хмарового сховища;
- відкритий вихідний код;
- легкий, високопродуктивний та модульний протокол HTTP-запитів.

Найкращою СУБД до цієї системи став MS SQL, який призначений для підтримки систем, що працюють на базі платформи .NET. За допомогою цієї бази та її інтеграції з .NET можна отримати найкращі показники розширюваності системи.

Також для швидкого та простого зв'язку між сервером та базою даних була використана об'єктно реляційний модуль зіставлення (ORM) Entity Framework (EF) Core, яка дозволяє розробникам .NET працювати з базою даних за допомогою об'єктів .NET і усуваючи необхідність в написанні здебільшого коду, необхідного для доступу до даних

4.2 Розробка прототипу системи


Як правило, робота над створенням сайту починається з оформлення прототипу проекту та його структури, це допоможе вирішити такі проблеми, як візуалізація ідей, представлення про зовнішній вигляд сайту на початковій стадії, можливість внесення змін, а також фіналізувати структуру проекту та отримати розуміння в кількості затраченого часу та можливості створення повторно використаних компонентів.

Основними блоками які потребують прототипи є :

- сторінка інформації про компанію з відгуками;
- сторінка створення запитання;
- сторінка моїх запитань
- форма перегляду запитання;
- форма призначення модераторів або відповідальних до системи;
- форма з найчастішими запитання.

Блок головної сторінки компанії де можна знайти основну інформацію про компанію та залишити відгук можна побачити на рисунку 4.1

					IT61.140БАК.004 ПЗ	Арк.
						47
Змн.	Арк.	№ докум.	Підпис	Дата		



Назва компанії

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

[Створити запитання](#)

[Основні телефони](#)

[Назва телефона]

[Номер телефона]

[Назва телефона]

[Номер телефона]

[Назва телефона]

[Номер телефона]

[Всі телефони](#)

Ваш відгук...

[ФІО]

[Текст відгуку]

Подобається Відповісти

[ФІО]

[Текст відгуку]

Подобається Відповісти

[ФІО]

[Текст відгуку]

Подобається Відповісти

[ФІО]

[Текст відгуку]

Подобається Відповісти

[ФІО]

[Текст відгуку]

Подобається Відповісти

[ФІО]

[Текст відгуку]

Подобається Відповісти

Ваш відгук...

[ФІО]

[Текст відгуку]

Подобається Відповісти

[ФІО]

[Текст відгуку]

Подобається Відповісти

Відправити

Дата

Дата

Дата

Дата

Відправити

Дата

Дата

5

Рисунок 4.1 – Макет сторінки корисної інформації з відгуками

Другим основним блоком системи є форма перегляду моїх запитань та створення запитань до компанії. Сторінка перегляду запитань, має функціональну кнопку, нове запитання, а також при натисканні на одне із запитань відкриває інформацію про нього. Блок сторінки запитань до компанії можна побачити на рисунку 4.2

Мої запитання

+

Нове запитання

Номер запитання	Дата створення	Тема
ID запитання	[Дата]	[Тема]
ID запитання	[Дата]	[Тема]
ID запитання	[Дата]	[Тема]

Рисунок 4.2 – Макет сторінки запитань користувача

При натисканні на функціональний елемент нове запитання відкривається сторінка створення запитання. Блок сторінки створення запитання до компанії можна побачити на рисунку 4.3

Форма створення запитань містить випадаючий список теми, які задає адміністратор чи модератор сайту. Містить перехід на список всіх запитань до цієї компанії, а також корисну інформацію яка знаходиться справа.

Запитання

Тема

[Тема]

Мої запитання →

Запитання

Введіть текст запитання

[Текст корисної інформації]

Відправити

Рисунок 4.3 – Макет сторінки створення запитання

При натисканні на запитання в списку запитань, відкривається форма перегляду запитання без відповіді, якщо її не було надано чи з відповіддю. Блок сторінки відображення запитання до компанії можна побачити на рисунку 4.4

✕

Запитання №[id]

Тема

[Тема]

Запитання

[Текст]

Рисунок 4.4 – Макет сторінки перегляду запитання без відповіді

Відображення запитання з відповіддю можна побачити на рисунку 4.4

✕

Запитання №[id]

Тема

[Тема]

Запитання

[Текст]

Відповідь

Автор

[ФІО]

Відповідь

[Текст]

Рисунок 4.5 – Макет сторінки перегляду запитання з відповіддю

Дати відповідь на запитання може адміністратор, модератор та відповідальний по темі, якому було назначено запитання.

Для того щоб назначити модератора на підприємство чи відповідального на тему існує форма для призначення. На сторінці списку відповідальних\модераторів. Пошук відповідального здійснюється по зареєстрованим користувачам сайту. Відображення призначення відповідального на тему компанії можна побачити на рисунку 4.6

Відповідальний

+ Додати

ФІО Відповідального

ФІО Ел.пошта

ФІО Ел.пошта

Додати відповідального

Відповідальний

Тема

Відмінити Додати

Рисунок 4.6 – Макет сторінки додавання відповідального на тему

До кожної теми може бути призначено декілька відповідальних. Модератор може відправляти запитання відповідальному, навіть якщо він не є відповідальним на темі запитання. Модератор, відповідальний та адміністратор, можуть переглядати всі запитання, а також давати відповідь. Відображення призначення модератора до компанії можна побачити на рисунку 4.7

Відповідальний

+ Додати

ФІО Модератора

ФІО Ел.пошта

ФІО Ел.пошта

Додати модератора

ФІО Модератора:

Відмінити Додати

Рисунок 4.7 – Макет сторінки додавання модератора

Якщо відповідальний зіткнувся з труднощами в наданні відповіді на питання, він може відправити його модератору, додавши відповідний підпис з причиною відправки. При відправці запитання модератору, запитання отримує статус «повернено» та відображає функціональний елемент «увага модератору», який відображається в списку усіх запитань модератора поруч з його ФІО. Система відправляє нотифікацію на пошту модератору, про те що відповідальний повернув йому запитання. В той же час модератор може відправити запитання відповідальному, який не є відповідальним на темі цього запитання. Відповідальний отримає нотифікацію на пошту про отримання запитання від модератора. Після надання відповіді відповідальним, модератор отримає нотифікацію, зможе переглянути відповідь відповідального та підтвердити її. Відображення призначення запитання відповідальному компанії можна побачити на рисунку 4.8

Рисунок 4.8 – Макет форми надання відповіді на запитання модератором

Відображення сторінки перегляду «Найчастіші запитання» можна побачити на рисунку 4.9

Найчастіші запитання

[Тема] ^

Можете будь-ласка відповісти на питання ? ^

Доброго дня, можу!

Питання
[ФІО Автора питання]

Відповідь
[ФІО Автора відповіді]

[Дата]

А де можна задати запитання. Запитання збільшене для того щоб показати перенос і розміщення великого запитання. v

А де можна задати запитання. Запитання збільшене для того щоб показати перенос і розміщення великого запитання.

Запитання звичайне, номер 3 v

А де можна задати запитання. Запитання збільшене для того щоб показати перенос і розміщення великого запитання. v

А де можна задати запитання. Запитання збільшене для того щоб показати перенос і розміщення великого запитання. v

А де можна задати запитання. Запитання збільшене для того щоб показати перенос і розміщення великого запитання. v

А де можна задати запитання. Запитання збільшене для того щоб показати перенос і розміщення великого запитання. v

А де можна задати запитання. Запитання збільшене для того щоб показати перенос і розміщення великого запитання. v

[Тема] v

[Тема] v

[Тема] v

Рисунок 4.9 – Макет сторінки з найчастішими запитаннями

Форма «Найчастіші запитання» розміщується на сторінці створення запитання, під формою створення запитання. Форма організована в вигляді відкриваючого блоку двох рівнів. Перший рівень, по темі запитання, другий рівень саме запитання. Кількість груп першого рівня дорівнює кількості тем які створить адміністратор для компанії.

					IT61.140БАК.004 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		53

4.3 Розробка структурної схеми бази даних

Архітектура СУБД програмного забезпечення є своєрідним планом системи і є первинним для розуміння. Створена структура СУБД має величезне значення для решти проекту і стає дуже важко змінити, чим більше ми просуваємося в процесі, тому дуже важливо підійти до цього відповідально. Саме структура бази даних дозволяє зробити систему розширюваною та легкою для реалізації.

Діаграму структури бази даних наведено у кресленику ІТ61.140БАК.004 ДЗ а її опис зазначено у таблиці 4.1

Таблиця 4.1 – Опис таблиці бази даних

Назва таблиці	Опис
dbo.user_credentials	Таблиця зберігає дані авторизації користувача, такі як: електронна пошта (mail varchar pk), пароль (password varchar)
dbo.users	Таблиця, що відповідає за користувачів системи. Має поля: ідентифікатор (userID uniqueidentifier pk), ім'я користувача (fullName varchar), пошта (mail varchar), посилання на аватарку користувача (logoUrl varchar).
dbo.companies	Таблиця, що відповідає за компанії системи. Має поля: ідентифікатор (companyID uniqueidentifier pk), ідентифікатор адміністратора компанії (userID uniqueidentifier fk), опис компанії (content varchar), посилання на картинку компанії (logoUrl varchar), назва компанії (companyName varchar).
dbo.feedbacks	Таблиця, що відповідає за відгуки системи. Має поля: ідентифікатор (feedbackID uniqueidentifier

Назва таблиці	Опис
	pk), ідентифікатор відгука на який було залишено відповідь (parentFeedbackID uniqueidentifier fk), текст відгука (content varchar), признак відредагованості відгука (isUpdated boolean), ідентифікатор компанії (companyID uniqueidentifier fk) , ідентифікатор користувача (userID uniqueidentifier fk), дата створення відгуку (creationDate DateTime)
dbo.questions	Таблиця, що відповідає за запитання системи. Має поля: ідентифікатор (questionID uniqueidentifier pk), ідентифікатор компанії (companyID uniqueidentifier fk), ідентифікатор користувача який створив запитання (userID uniqueidentifier fk), ідентифікатор теми до якої було поставлене запитання (topicID uniqueidentifier fk), признак чи питання є найчастішим (isFAQ boolean) , повідомлення про повернення питання модератору (moderatorAlarm boolean), причина призначення запитання (reasonForSending varchar), текст запитання (text varchar) , дата створення запитання (creationDate DateTime), ідентифікатор відповіді на запитання (questionAnswerID uuid fk).
dbo.answers	Таблиця, що відповідає за відповіді на запитання. Має поля: ідентифікатор (answerID uuid pk), ідентифікатор користувача який надав відповідь (userId uniqueidentifier fk), текст відповіді (text varchar), дата створення відповіді (createdDate DateTime).
dbo.topics	Таблиця, що відповідає за теми запитань. Має поля: ідентифікатор (topicID uniqueidentifier pk), ідентифікатор користувача який є відповідальним за

Назва таблиці	Опис
	тему (responsibleID uniqueidentifier fk), назва теми (topicName varchar), опис теми (topicExplanation varchar).
dbo.user_company	Єднальна таблиця, що зв'язує таблицю користувачів з таблицею компаній. Містить у собі власний ідентифікатор (id uuid pk), ідентифікатори користувача (userID uniqueidentifier fk), ідентифікатор компанії (companyID uniqueidentifier fk), признак модератора (isUserModerator boolean), признак збереженої компанії (isUserLiked boolean), признак адміністратора (isUserAdmin boolean).
dbo.likes	Таблиця, що відповідає за оцінку відгуків. Має поля: ідентифікатор (likeID uniqueidentifier pk), ідентифікатор користувача який оцінив відгук (userID uniqueidentifier fk), ідентифікатор відгука, який був оцінений (feedbackID uniqueidentifier fk).
dbo.hotlines	Таблиця, що відповідає за гарячі лінії компанії. Має поля: ідентифікатор (hotlineID uniqueidentifier pk), ідентифікатор компанії (companyID uniqueidentifier fk), назва гарячої лінії (hotlineName varchar), номер гарячої лінії (hotlineNumber varchar).
__EFMigrations	Таблиця, що містить інформацію про міграції застосунку – набір змін моделей EntityFramework, яка створює необхідні команди для зміни таблиць бази даних. Має поля: власний ідентифікатор міграції (id uuid pk), назва міграції (migrationName varchar), дата та час застосування міграції (migrationDate DateTime).

4.4 Структура проекту

Структуру проекту можна побачити на рисунку 4.10

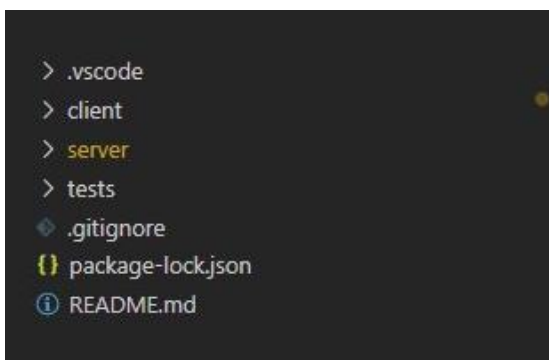


Рисунок 4.10 – Структура проекту

- .vscode – папка яка призначена для налаштування середовища розробки;
- client – папка яка включає в себе клієнтський модуль проекту;
- package-lock.json– файл який містить всі пакети для підтримки проекту;
- server – папка яка включає в себе серверний модуль проекту;
- tests – папка, яка включає в себе модуль тестування проекту;
- .gitignore – налаштовує зв'язок з зовнішнім репозиторієм проекту;
- Readmy.md– опис програми.

4.4.1 Структура серверної частини

Структуру серверної частини проекту можна побачити на рисунку 4.11

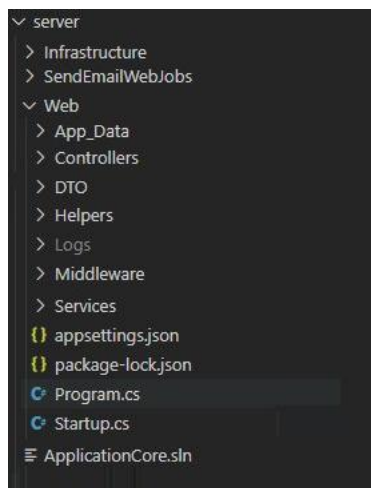


Рисунок 4.11 – Структура серверної частини

					IT61.140БАК.004 ПЗ	Арк.
						57
Змн.	Арк.	№ докум.	Підпис	Дата		

- Infrastructure – папка яка відповідає за взаємодію бази даних з сервером, міграції системи, а також за контекст бази даних;
- SendEmailWebJob – папка яка містить SQL роботу, яку повинна виконувати база даних для того, щоб відправляти нотифікації на пошту, або для виконання щоденних задач;
- Web – папка яка є кореневою, здійснює всю серверну логіку, наставляє повноцінний MVC паттерн та містить допоміжні системні блоки:
 - а) App_Data – папка яка містить модель системи (класи), які є складовою патерна MVC і головною частиною проекту;
 - б) Controllers – папка яка містить контролери системи, які відповідають за взаємодію між інтерфейсом та моделлю;
 - в) DTO – папка яка містить модель шаблону data transfer object, яка потрібна для безпечної передачі даних між клієнтом та сервером;
 - г) Helpers – папка яка містить допоміжні функції системи, такі як трансформатори дати в правильний формат, перетворення класів;
 - г) Logs – папка яка містить логи системи, для звітування роботи системи та її модулів;
 - д) Middleware – папка яка містить проміжне програмне забезпечення яке допомагає взаємодіяти між модулями системи;
 - е) Services – папка яка містить сервіси для виконання функцій коду;
 - є) Appsettings.json – файл який відповідає за наставлявання змінних серверного модулю;
 - ж) Package-lock.json – файл який містить всі пакети для підтримки сервера;
 - з) Program.cs – файл який є точкою входу запуску серверної частини системи;
 - и) Startup.cs – файл який є точкою з'єднання всіх блоків системи та задає цим блокам параметри;
- ApplicationCore.sln – файл, який налаштовує серверну частину проекту.

4.4.2 Структура клієнтської частини

Структуру клієнтської частини проекту можна побачити на рисунку 4.12

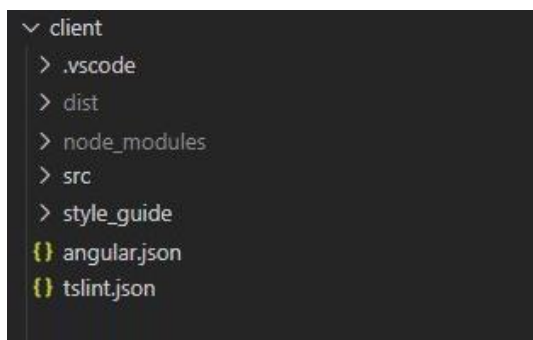


Рисунок 4.12 – Структура клієнтської частини

- .vscode – папка яка призначена для налаштування середовища розробки;
- dist – папка яка призначена для фінальної продуктивної версії проекту;
- node_modules – папка яка містить сторонні бібліотеки потрібні для підтримки проекту;
- src – коренева папка клієнтської сторони, яка містить фронт-енд код, розмітку, компоненти, стилі;
- style_guide – папка яка містить правила написання коду для розробки клієнтської сторони;
- angular.json – основний файл клієнтської частини, який її налаштовує;
- tslint.json – файл який відповідає за допоміжний синтаксис при розробці, та вказує на помилки, ще до етапу компіляції.

4.4.3 Структура тестової частини

Структуру тестової частини можна побачити на рисунку 4.13

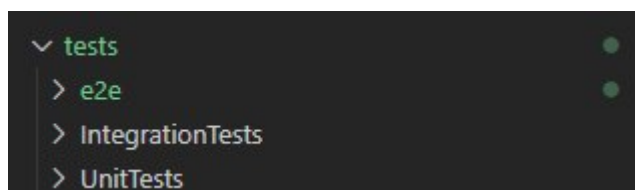


Рисунок 4.13 – Структура тестової частини

- e2e – папка з «end-to-end» тестами проекту. Цілю тестування «end-to-end» є моделювання реального сценарію користувача та перевірка інтерфейсу проекту на правильність відображення;
- integrationTests – папка з інтеграційними тестами проекту. Цілю тестування є перевірка взаємодії між компонентами системи;
- unitTests – папка з модульними тестами проекту. Цілю тестування є перевірка на коректність окремих блоків коду проекту.

4.5 Реалізація системи

Дана система складається з компонентів серверної та клієнтської сторони. Відношення між програмними модулями зображено на кресленику IT61.140БАК.004 Д4.

4.5.1 Автентифікація

Одною із головних функціональних особливостей даної системи є рольова модель, а отже є необхідність реалізувати автентифікацію користувача. Головним архітектурним шаблоном є model-view-controller, який включає в себе принципи REST і є незалежний від стану (stateless), тому клієнтська сторона повинна сама подбати про свою автентифікацію при кожному запиті. Саме тому було використано технологію - JSON Web Token або просто JWT.

Технічно JWT являє собою веб-стандарт, який визначає спосіб передачі даних про користувача в форматі JSON в зашифрованому вигляді.[11]

Після заповнення коректного логіну та паролю на формі авторизації, клієнту повертається згенерований сервером JWT. З кожним наступним запитом разом передається JWT, сервер декодує його і перевіряє на валідність, а також чи термін дії токена не був вичерпаний. Якщо все в порядку запит відправляє відповідь.

					IT61.140БАК.004 ПЗ	Арк.
						60
Змн.	Арк.	№ докум.	Підпис	Дата		

Діаграму послідовності процесу авторизації через JWT токен зображено на рисунку 4.14

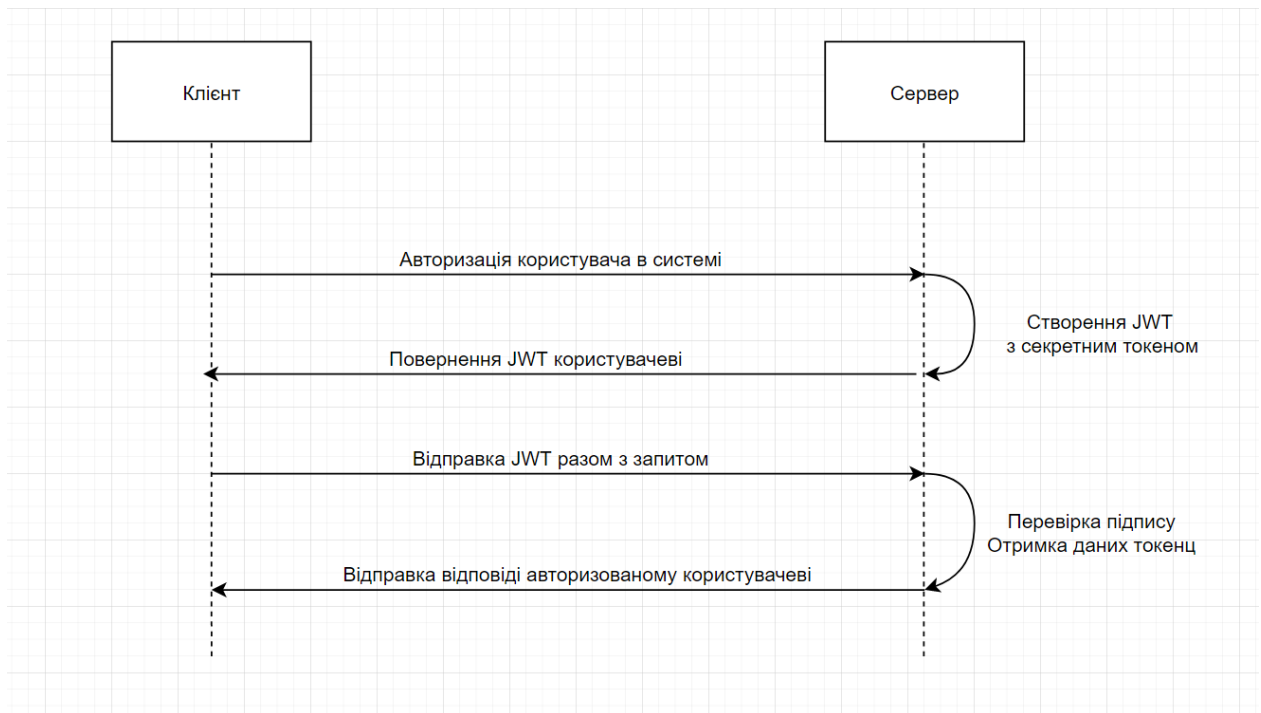


Рисунок 4.14 – Діаграма послідовності авторизації

Форма авторизації зображена на рисунку 4.15

Увійти

Електронна пошта

Введіть пароль

.....|

[Способи входу](#)

Назад

Далі

Рисунок 4.15 – Форма авторизації

Процес автентифікації також включає процес реєстрації в системі. Реєстрація здійснюється в декілька етапів:

- 1) заповнення даних в формі інтерфейсу. Форма реєстрації зображена на рисунку 4.16;
- 2) перевірка на коректність вводу та відправка моделі через запит;
- 3) обробка та перевірка на валідність запиту контролером.

Створення облікового запису

Введіть вашу дійсну електронну пошту

abc@example.com

Введіть пароль, який ви хочете використовувати з обліковим записом.

Створіть пароль

Введіть ваше ФІО

Введіть ваше повне ім'я

Зареєструватися

Рисунок 4.16 – Форма реєстрації

В реалізації реєстрації був використаний паттерн data transfer object (DTO).

DTO - це клас, що містить дані без якої-небудь логіки для роботи з ними. Шаблон використовуються для передачі даних між різними слоями всередині однієї програми.[12]

У таблиці 4.2 наведено опис моделі, що відповідає за реєстрацію користувача.

					IT61.140БАК.004 ПЗ	Арк.
						62
Змн.	Арк.	№ докум.	Підпис	Дата		

Таблиця 4.2 – Опис моделі, що відповідає за реєстрацію користувача

Назва класу	Поля
UserDTO	<p>Модель користувача має наступні поля:</p> <ul style="list-style-type: none"> - FullName – ім'я користувача. Має обмеження на довжину в 128 символів; - Email – пошта користувача, має бути записаною в форматі електронної пошти; - Password – пароль до облікового запису. Має бути не меншим ніж 8 символів.

4.5.2 Адміністрування компанії

Адміністративний блок призначений для швидкого налаштування системи під свою компанію. Він передбачає:

- управління компанією;
- управління гарячими лініями;
- управління модераторами;
- управління відповідальними.

У таблиці 4.3 наведено опис моделі, що відповідає за створення компанії.

Таблиця 4.3 – Опис моделі, що відповідає за управління компанією

Назва класу	Поля
CompanyDTO	<p>Модель компанії має наступні поля:</p> <ul style="list-style-type: none"> - FullName – назва компанії не більше 128 символів; - Description – опис компанії не більше 516 символів; - LogoCompany – відповідає за мініатюру зображення компанії, якщо зображення не обрано,

	<p>встановлює за замовчуванням зображення «images/empty.jpg»;</p> <p>- Topics – відповідає за теми для запитань в компанії.</p>
--	---

Опис видів управління компанії наведено у таблицях 4.4-4.8

Таблиця 4.4 – Опис виду сторінки моїх та збережених компаній

Назва методу	GetAllCompanies
Опис роботи	<p>Вид відправляє запит на контролер системи CompanyController, який отримує з бази даних усі компанії створені користувачем, а також усі компанії які користувач зберіг. Потім отримує відповідь та відображає у вигляді двох списків.</p>

Відображення списків компаній зображено на рисунку 4.17

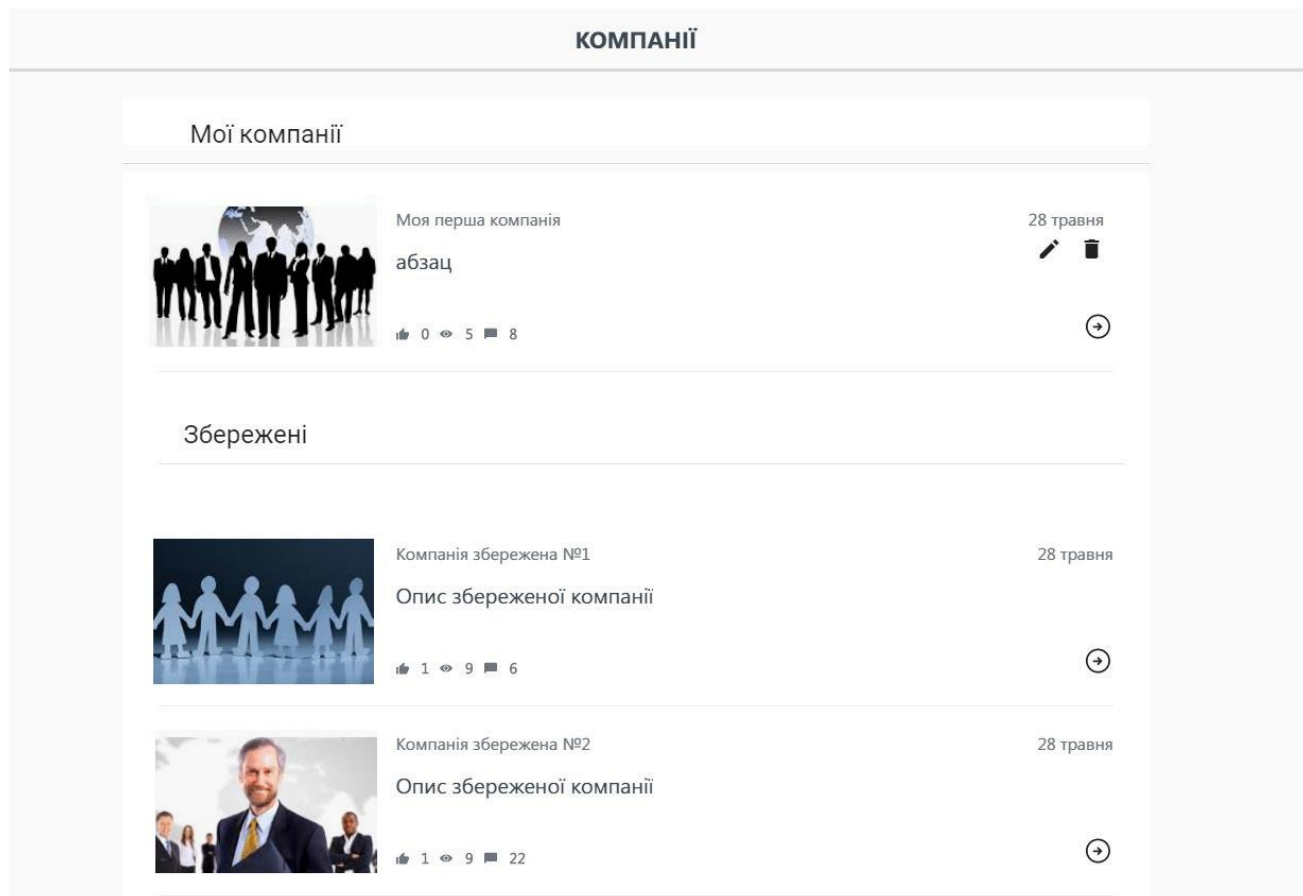


Рисунок 4.17 – Відображення списків компаній

Таблиця 4.5 – Опис виду створення нової компанії

Назва методу	CreateCompany
Опис роботи	При перегляді виду доступна форма з полями для вводу, а також кнопка «Підтвердити». Після заповнення обов'язкових полів форми та натисканні на кнопку, поля проходять процедуру валідації. Якщо дані не проходять перевірку, то генерується повідомлення з помилкою під полями які не пройшли валідацію. При успішному результаті відправляється «POST» запит до контролера CompanyController, який записує компанію до бази даних.

Форма для створення нової компанії зображена на рисунку 4.18

Таблиця 4.6 – Опис виду редагування нової компанії

Назва методу	UpdateCompany
Опис роботи	При перегляді виду редагування компанії, відкривається форма створення компанії. Далі відправляється запит «GET», який отримує дані про компанію та заповнює поля форми. Після зміни значень полів форми та натисканні на кнопку, поля проходять процедуру валідації. Якщо дані не проходять перевірку, то генерується повідомлення з помилкою під полями які не пройшли валідацію. При успішному результаті, поля які змінилися відправляються через «PUT» запит до контролера CompanyController, який змінює інформацію про компанію в базі даних.

Форма редагування компанії зображена на рисунку 4.18

НОВА КОМПАНІЯ

Нова компанія

Ваша компанія

Додати зображення

File Edit View Insert Format Tools Table

Paragraph B I S A Link Text Color Background Color

Опис компанії

Р 2 WORDS POWERED BY TINY

Теми до запитань

☒ Додати тему

Головна тема запитання

Рисунок 4.18 – Форма створення компанії

Таблиця 4.7 – Опис виду перегляду інформації про компанію

Назва методу	GetCompanyById
Опис роботи	<p>Вид відправляє запит «GET» на контролер системи CompanyController, який отримує з бази даних конкретну</p> <p>Компанію. Потім отримує відповідь та відображає компанію.</p>

Відображення інформації про компанію зображено на рисунку 4.19

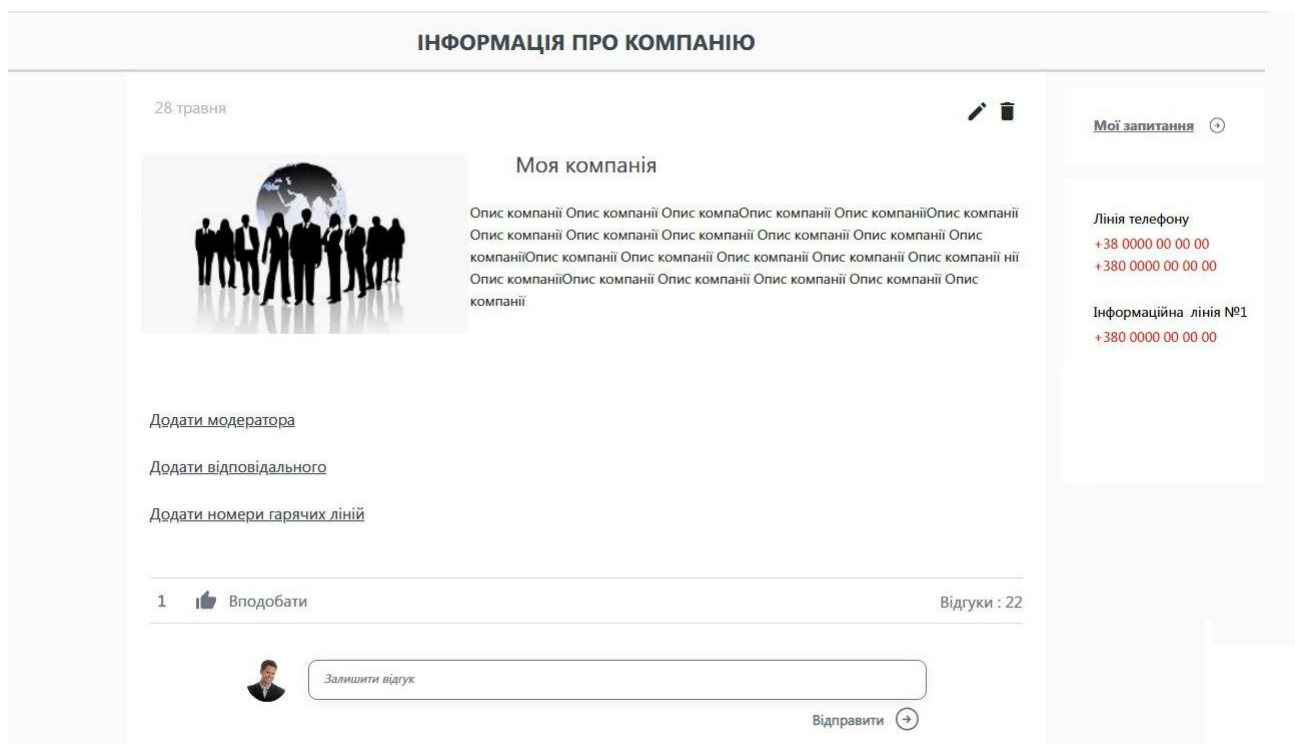


Рисунок 4.19 – Відображення інформації про компанію

Таблиця 4.8 – Опис виду видалення компанії

Назва методу	DeleteCompany
Опис роботи	При натисканні функціонального елемента видалити на формі перегляду списків компаній або на формі перегляду компанії відображається діалогове вікно з підтвердженням дії. Якщо підтвердити дію, то відправляється запит «Delete» до контролера CompanyController, який видаляє компанію з бази даних, а також каскадно видаляє всі дані які були пов'язані з компанією (відгуки, запитання).
	Потім повертає відповідь та переходить на сторінку перегляду моїх та збережених компаній.

Діалогове вікно підтвердження видалення компанії зображено на рисунку 4.20

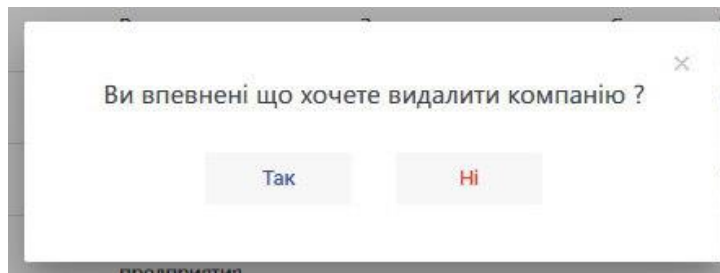


Рисунок 4.20 – Діалогове вікно підтвердження видалення компанії

У таблиці 4.9 наведено опис моделі, що відповідає за створення компанії.

Таблиця 4.9 – опис моделі, що відповідає за управління компанією

Назва класу	Поля
HotlineDTO	<p>Модель гарячої лінії має наступні поля:</p> <ul style="list-style-type: none"> - HotlineName – назва гарячої лінії не більше 128 символів; - HotNumber – номер лінії не більше 30 цифр; - CompanyId – ідентифікатор компанії якій належить лінія.

Опис видів управління гарячими лініями наведено у таблицях 4.10 - 4.12

Таблиця 4.10 – Опис виду списку гарячих ліній компанії

Назва методу	GetCompanyHotlines
Опис роботи	<p>Вид відправляє запит на контролер системи HotlineController, який отримує з бази даних усі гарячі лінії компанії. Потім отримує відповідь контролера та відображає список гарячих ліній.</p>

Відображення списку гарячих ліній зображено на рисунку 4.21



ГАРЯЧІ ЛІНІЇ			
+ Додати		Оновити	
	№ ↓	Номер	Назва
	1	0 800 000 000 00	Гаряча лінія №1
	2	0 800 000 000 00	Гаряча лінія №1
	3	0 800 000 000 00	Гаряча лінія №1
	4	0 800 000 000 00	Гаряча лінія №1
	5	0 800 000 000 00	Гаряча лінія №1
	6	0 800 000 000 00	Гаряча лінія №1

Рисунок 4.21 – Відображення списку гарячих ліній

Таблиця 4.11 – Опис виду створення гарячої лінії

Назва методу	CreateHotline
Опис роботи	При натисканні на функціональний елемент «Додати» на сторінці переглянута всіх гарячих ліній, відкривається діалогове вікно з полями для вводу, а також кнопка «Відправити». Після заповнення обов’язкових полів форми та натисканні на кнопку, поля проходять процедуру валідації. Якщо дані не проходять перевірку, то генерується повідомлення з помилкою під полями які не пройшли валідацію. При успішному результаті відправляється «POST» запит до контролеру HotlineControler, який записує гарячу лінію до бази даних.

Форма для створення нової гарячої лінії зображена на рисунку 4.22

Додати гарячу лінію

Лінія

Номер

Відміна Відправити

Рисунок 4.22 – Форма створення гарячої лінії

Таблиця 4.12 – Опис виду видалення гарячої лінії

Назва методу	DeleteHotline
Опис роботи	При натисканні функціонального елементу видалити на формі перегляду гарячих ліній компаній відображається діалогове вікно з підтвердженням дії. Якщо підтвердити дію, то відправляється запит «Delete» до контролера HotlineController, який видаляє гарячу лінію з бази даних. Потім повертає відповідь та видаляє лінію із списку гарячих ліній.

Діалогове вікно підтвердження видалення компанії зображено на рисунку 4.23

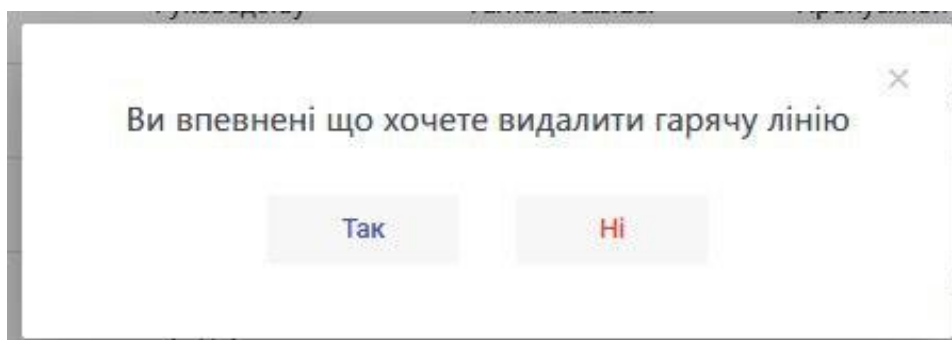


Рисунок 4.23 – Діалогове вікно підтвердження видалення гарячої лінії

Таблиця 4.13 – опис моделі, управління модераторами компанії

Назва класу	Поля
ModeratorDTO	<p>Модель модератора має наступні поля:</p> <ul style="list-style-type: none"> - UserId – ідентифікатор користувача якого потрібно призначити модератором; - CompanyId – ідентифікатор компанії;

Опис видів управління модераторами наведено у таблицях 4.14-4.16

Таблиця 4.14 – Опис виду сторінки модераторів компаній

Назва методу	GetAllModerators
Опис роботи	Вид відправляє запит «GET» на контролер системи ModeratorController, який отримує з бази даних усіх модераторів компанії. Потім отримує відповідь контролера та відображає список модераторів.

Відображення списку модераторів зображено на рисунку 4.24

МОДЕРАТОРИ				
+ Додати		Обновити		
		Id ↓	Модератор	
		1	Іванов Іван Іванович	
		2	Борисов Борис Борисович	
		3	Іваненко Іван Іванович	
		4	Костенко Ліна Іванівна	
		5	Іванов Іванчик Іванович	
		6	Івасик Валентин Іванович	

Рисунок 4.24 – Відображення модераторів компаній

Таблиця 4.15 – Опис виду створення модератора компанії

Назва методу	CreateModerator
Опис роботи	<p>При натисканні на функціональний елемент «Додати» на сторінці перегляду всіх модераторів, відкривається діалогове вікно з полем для вибору модератора, а також кнопка «Відправити». Після заповнення обов’язкового поля форми та натисканні на кнопку, поле проходить процедуру валідації. Якщо дані не проходять перевірку, то генерується повідомлення з помилкою під полем.</p> <p>При успішному результаті відправляється «POST» запит до контролера ModeratorControler, який записує модератора до бази даних, та зв’язує модератора та компанію.</p>

Форма для створення нової компанії зображена на рисунку 4.25

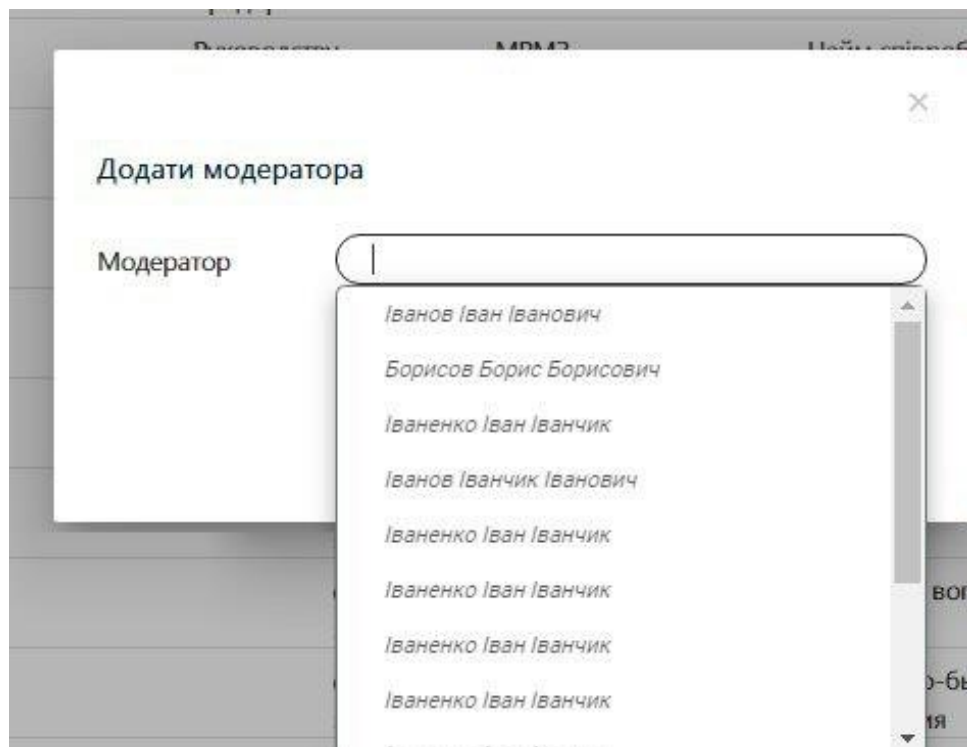


Рисунок 4.25 – Відображення форми додавання модератора

Таблиця 4.16 – Опис виду видалення модератора

Назва методу	DeleteModerator
Опис роботи	При натисканні функціонального елемента видалити на формі перегляду модераторів компаній, відображається діалогове вікно з підтвердженням дії. Якщо підтвердити дію, то відправляється запит «Delete» до контролера ModeratorController, який видаляє модератора з бази даних. Потім повертає відповідь та видаляє модератора із списку.

Діалогове вікно підтвердження видалення компанії зображено на рисунку 4.26

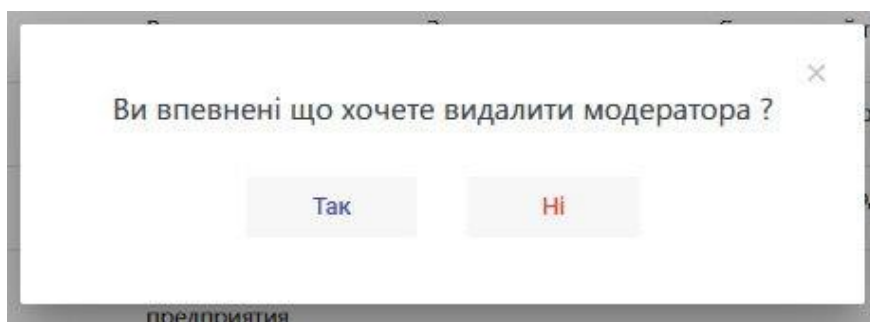


Рисунок 4.26 – Відображення діалогового вікна видалення модератора

У таблиці 4.17 наведено опис моделі, що відповідає за призначення відповідального за тему.

Таблиця 4.17 – опис моделі, що відповідає за управління відповідальними за тему компанії.

Назва класу	Поля
ResponsibleDTO	<p>Модель модератора має наступні поля:</p> <ul style="list-style-type: none"> - UserId – ідентифікатор користувача якого потрібно призначити модератором; - TopicId – ідентифікатор теми;

Опис видів управління відповідальними наведено у таблицях 4.18-4.20

Таблиця 4.18 – Опис виду сторінки модераторів компанії

Назва методу	GetAllResponsible
Опис роботи	Вид відправляє запит «GET» на контролер системи ResponsibleController, який отримує з бази даних усіх відповідальних компанії. Потім отримує відповідь контролера та відображає список відповідальних.

Відображення списку відповідальних зображено на рисунку 4.27

ВІДПОВІДАЛЬНІ			
<div> <div>+ Додати</div> <div>Обновити</div> </div>			
	Id ↓	Тема	Відповідальний
	1	Тех ремонт	Іванов Іван Іванович
	2	Транспорт	Борисов Борис Борисович
	3	Офіс	Іваненко Іван Іванович
	4	Найм співробітників	Костенко Ліна Іванівна
	5	Склад	Іванов Іван Іванович
	6	Медична допомога	Івасик Валентин Іванович

Рисунок 4.27 – Відображення відповідальних компаній

Таблиця 4.19 – Опис виду створення модератора компанії

Назва методу	CreateResponsible
Опис роботи	<p>При натисканні на функціональний елемент «Додати» на сторінці перегляду всіх відповідальних, відкривається діалогове вікно з полем для вибору відповідального та тему на яку його призначити, а також кнопка «Відправити». Після заповнення обов’язкового поля форми та натисканні на кнопку, поле проходить процедуру валідації. Якщо дані не проходять перевірку, то генерується повідомлення з помилкою під полем. При успішному результаті відправляється «POST» запит до контролеру ResponsibleControler, який записує відповідального до бази даних, та зв’язує відповідального та тему.</p>

Форма для створення нової компанії зображена на рисунку 4.28

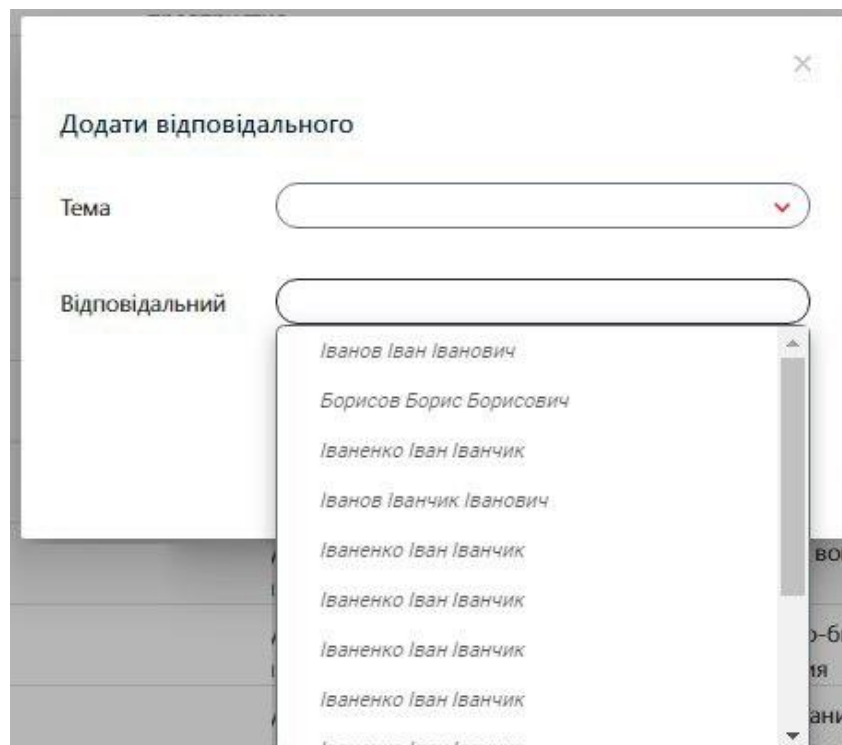


Рисунок 4.28 – Відображення форми додавання відповідального

Таблиця 4.21 – Опис виду видалення відповідального

Назва методу	DeleteResponsible
Опис роботи	При натисканні функціонального елемента видалити на формі перегляду відповідального компаній, відображається діалогове вікно з підтвердженням дії. Якщо підтвердити дію, то відправляється запит «Delete» до контролера ResponsibleController, видаляє відповідального з бази даних. Потім повертає відповідь та видаляє відповідального із списку.

Діалогове вікно підтвердження видалення компанії зображено на рисунку 4.29

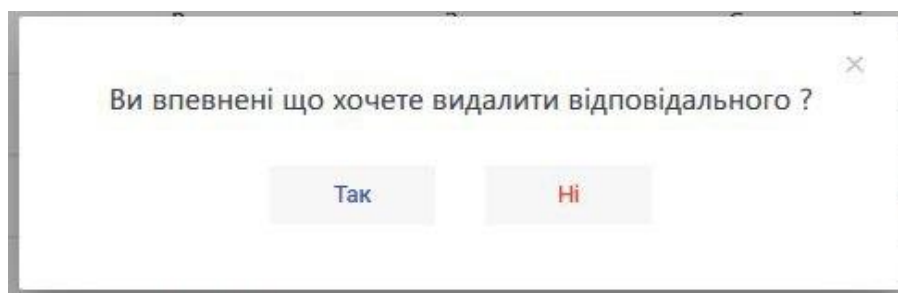


Рисунок 4.29 – Відображення діалогового вікна видалення відповідального

4.5.3 Відгуки

Блок відгуків призначений для залишення враження про компанію. Він передбачає:

- перегляд відгуків;
- створення відгуку;
- редагування відгуку;
- оцінювання відгуку;
- відповіді на відгук.

У таблиці 4.22 наведено опис моделі, що відповідає за створення компанії.

Таблиця 4.22 – Опис моделі, що відповідає за управління компанією

Назва класу	Поля
FeedbackDTO	<p>Модель відгуку має наступні поля:</p> <ul style="list-style-type: none"> - UserId – ідентифікатор користувача який залишив відгук; - CompanyId – ідентифікатор компанії в якій залишили відгук; - Content – текст відгуку, не більше 516 символів; - ParentFeedbackId – відповідає за ідентифікатор відгука на який було надано відповідь; - CreatedDate – відповідає за час створення відгуку типу DateTime; - EditedDate – відповідає за час редагування відгуку.

Опис виду управління наведено у таблицях 4.23-4.27

Таблиця 4.23 – Опис виду сторінки перегляду всіх відгуків

Назва методу	GetAllFeedbacks
Опис роботи	Вид відправляє запит на контролер системи FeedbackController, який отримує з бази даних усі відгуки залишені в компанії. Потім вид отримує відповідь та відображає список відгуків.

Відображення списку відгуків зображено на рисунку 4.30

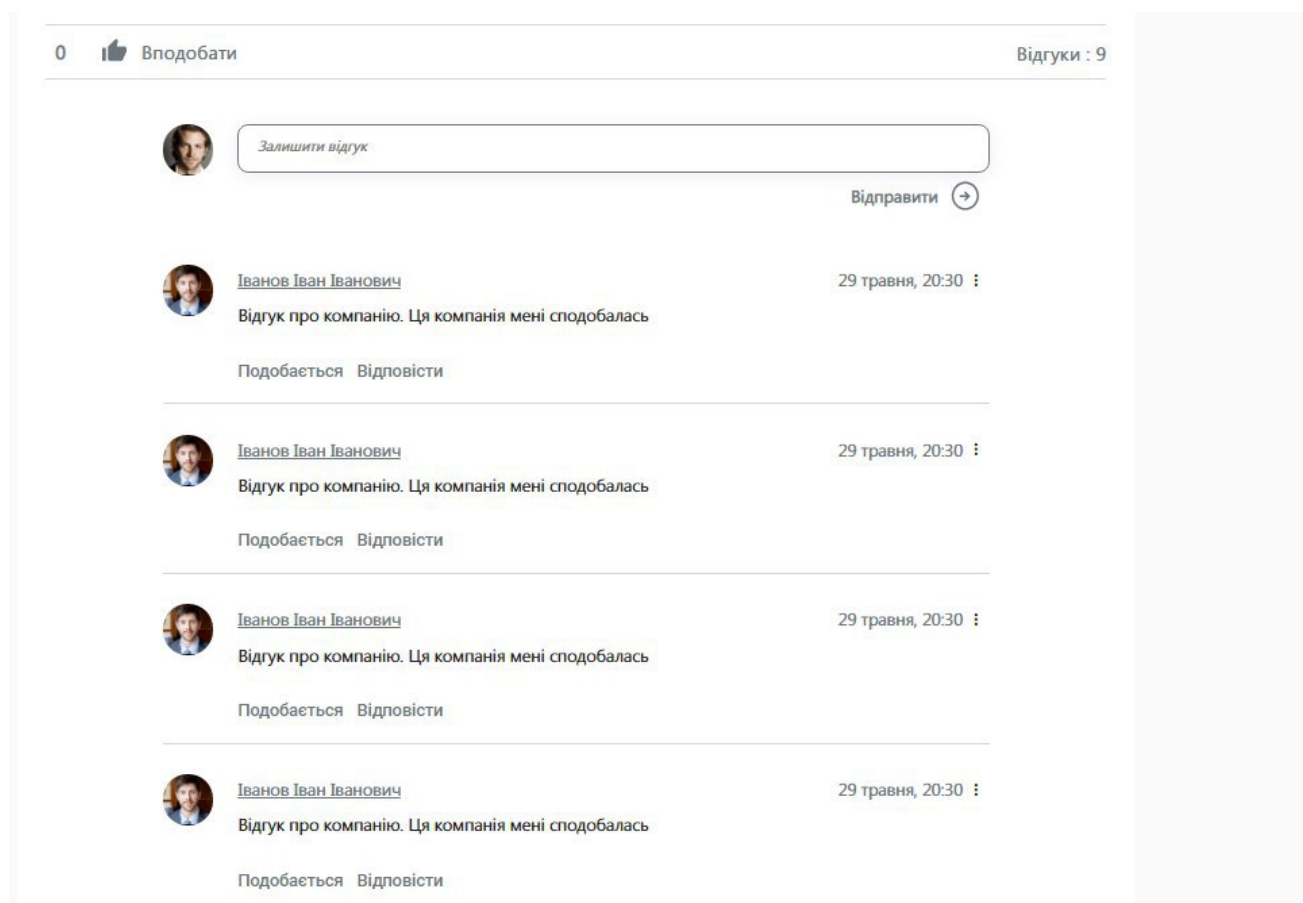


Рисунок 4.30 – Відображення списку відгуків компаній

Таблиця 4.24 – Опис виду створення нового відгуку

Назва методу	CreateFeedback
Опис роботи	При перегляді виду інформації для компанії доступне поле для вводу, а також кнопка «Відправити». Після заповнення поля та натисканні на кнопку відправляється «POST» запит до контролеру FeedbackController, який записує відгук до бази даних та створює зв'язок між компанією та відгуком. Повертає відповідь про успішне додавання відгуку та додає його до списку відгуків

Форма для додавання нового відгуку зображене на рисунку 4.31



Рисунок 4.31 – Форма для додавання нового відгуку

Таблиця 4.25 – Опис виду редагування відгуку

Назва методу	EditFeedback
Опис роботи	При натисканні на функціональний елемент «редагувати» біля залишеного відгуку, відкривається форма для редагування відгуку. При зміні тексту відгука та натисканні кнопки «Відправити», посилається запит «PUT» запит до контролеру FeedbackController, який змінює інформацію про відгук в базі даних, після чого відправляє відповідь, змінює відгук в списку вігуків та додає функціональний елемент «відредагований» до відгука

Вигляд зі зміненим відгуком зображений на рисунку 4.32



Рисунок 4.32 – Форма для додавання нового відгуку

Таблиця 4.26 – Опис виду відповіді на відгук

Назва методу	CreateFeedback
Опис роботи	На вигляді списку відгуків, доступна кнопка «Відповісти», при натисканні на яку відображається форма створення відгуку а також кнопка «Відправити». Після заповнення поля та натисканні на кнопку «Відправити» відправляється «POST» запит, до контролеру FeedbackController, який записує відгук до бази даних та створює зв'язок між відгуком до якого було надано відповідь та самим відгуком. Повертає відповідь про успішне додавання відгуку та додає його до списку відповідей на відгук. Якщо відповідей більше двох, також додає функціональний елемент «Приховати відповіді», який приховує список відповідей на відгук.

Відображення списку з формою для залишення відповіді, а також зі списком відгуків конкретної компанії зображено на рисунку 4.33

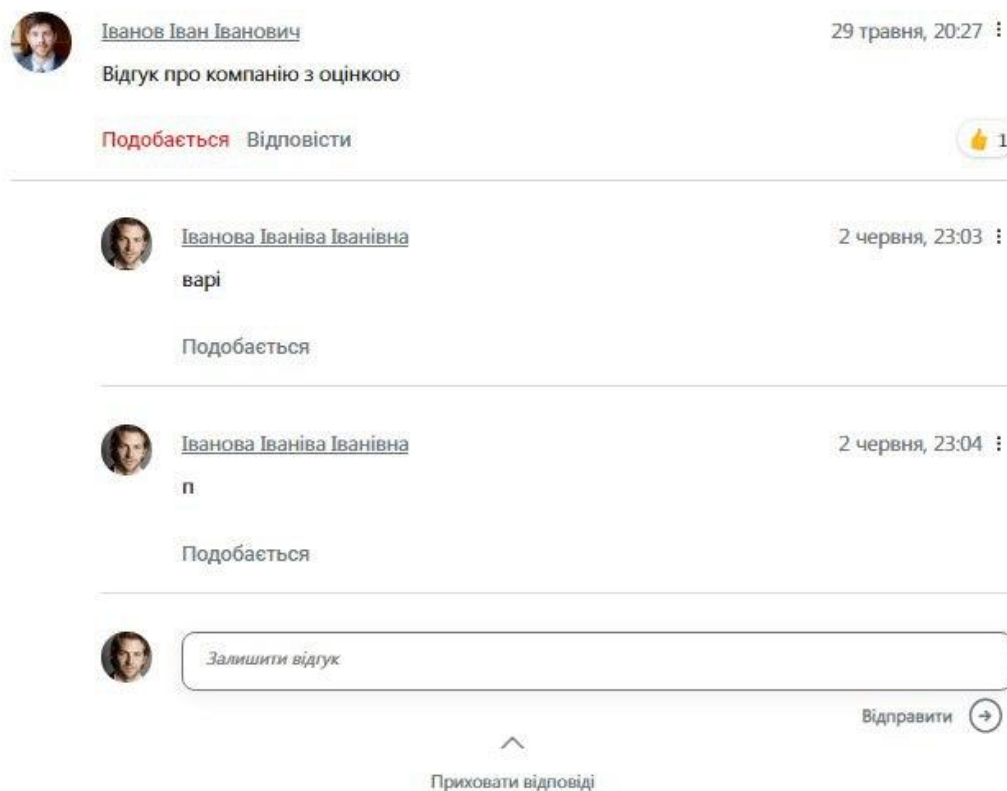


Рисунок 4.33 – Відображення списку відповідей на відгук

Таблиця 4.27 – Опис виду оцінки відгуку

Назва методу	LikeFeedback
Опис роботи	<p>При перегляді списку відгуків та списку відповідей на відгук дотупний функціональний елемент «Подобається». При натисканні на нього відправляється «POST» запит до контролеру LikeController, який записує оцінку до бази даних та створює зв'язок між оцінкою та відгуком. Повертає відповідь про успішне додавання оцінки та додає його до відгуку у вигляді функціонального елемента «подобається» та відображає кількість вподобань. Також підсвічує вже оцінений відгук.</p> <p>Якщо відгук вже оцінений, то при натисканні на функціональний елемент подобається відправляється</p>

	«Delete» запит до контролеру LikeController, який видаляє оцінку з бази даних, повертає успішний результат та знімає підсвічення, а також зменшує кількість вподобань на 1.
--	---

Відображення оціненого відгуку зображено на рисунку 4.34

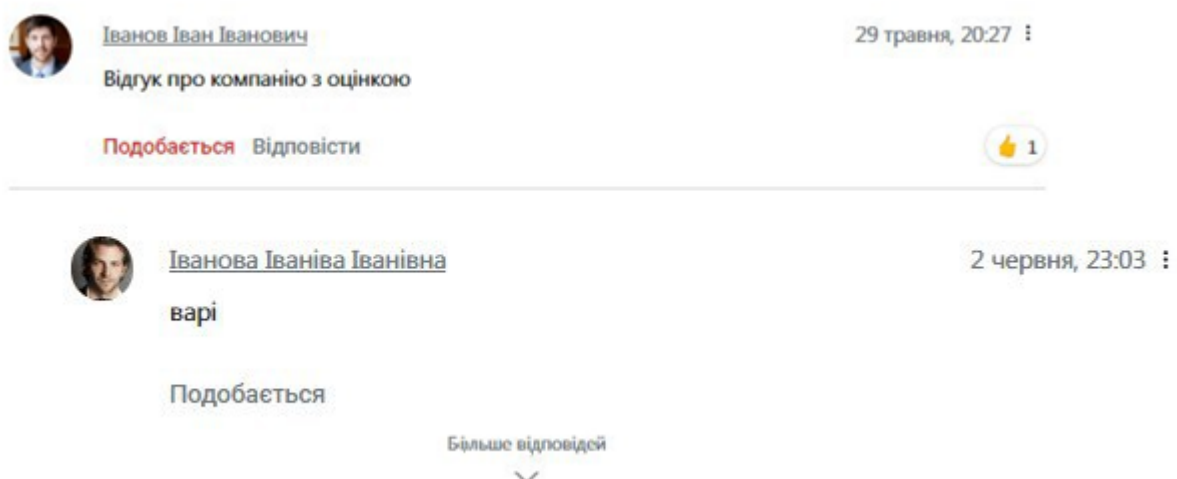


Рисунок 4.34 – Відображення оціненого відгуку

Висновки до розділу

У цьому розділі були оглянуті технології які забезпечать можливість дотримання усіх функціональних та нефункціональних вимог, описано структуру проекту, яка включає в себе клієнтський модуль, серверний модуль та модуль тестування. Було розроблено прототип системи та структуру бази даної. Було реалізовано серверну сторону та інтерфейс користувача. Було використано паттерн MVC(model-view-controller) для розширюваної архітектури та паттерн DTO(data-transfer-object) для безпечного передавання даних між частинами системи.

					IT61.140БАК.004 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		82

5 ТЕСТУВАННЯ СИСТЕМИ ЗВОРОТНОГО ЗВ'ЯЗКУ

5.1 Доцільність тестування

Тестування програмного забезпечення є важливою частиною в розробці. Воно має багато переваг. Одна з найважливіших - економічна ефективність. Тестування у вашому проекті може заощадити час в перспективі. Розробка програмного забезпечення складається з багатьох етапів і якщо помилки потрапляють на більш ранніх етапах, їх виправити займе набагато менше часу.

Мій продукт створений для публічного обслуговування користувачів, тому дуже важливо, щоб він приносив цінність, яку він обіцяє, отже, він повинен працювати належним чином, щоб забезпечити чудовий відгук від клієнтів. Розробка додатку включає багато процесів і тестування дає змогу перевірити практично всі аспекти. Тестування перевіряє правильність вирівнювання графіки програм, тестує основні функціональні можливості, перевіряє чи меню інтуїтивно зрозуміло тощо.

Існує три види тестування:

- unit-тестування;
- Інтеграційне тестування;
- тестування «end-to-end».

5.2 Unit-тестування

Unit-тестування - тип тестування, щоб перевірити, чи маленький фрагмент коду працює коректно.[13]

Unit тестування перевіряє один компонент програми. Кожному тесту повинен відповідати окремий сценарій. Область тестування Unit є вузькою, вона охоплює блок або невеликий фрагмент коду, що перевіряється, тому під час написання одиничного тесту використовуються більш короткі коди, які орієнтуються лише на один клас. Unit тести не повинні мати залежності від коду поза тестованою одиницею. Цей тип тестування повинен проводитися в

					IT61.140БАК.004 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		83

життєвому циклі тестування програмного забезпечення та, як правило, виконуватись розробником.

Декілька опрацьованих сценаріїв, тестові випадки та їх результати є наведеними у таблицях 5.1-5.5.

Таблиця 5.1 – Тестовий випадок «Відображення форми авторизації»

Функція	Відображення форми авторизації.
Результат функції	- перехід на форму авторизації; - форма відкрилася; - форма містить два поля.
Результат тесту	Пройшов.

Таблиця 5.2 – Тестовий випадок «Авторизація»

Функція	Авторизація
Результат функції	- форма авторизації відкривається; - не ввести дані та натиснути «Авторизуватися»; - з'являється повідомлення про помилку; - авторизація не відбувається.
Результат тесту	Пройшов.

Таблиця 5.3 – Тестовий випадок «Оцінювання відгуку»

Функція	Оцінка неоціненого відгуку
Результат функції	- кількість оцінок відгуку збільшено; - кнопка «подобається» підсвічена червоним; - додано функціональний елемент «подобається» до відгуку.
Результат тесту	Пройшов.

Таблиця 5.4 – Тестовий випадок «Оцінка оціненого відгуку»

Функція	Оцінка оціненого відгука
Результат функції	<ul style="list-style-type: none"> - кількість оцінок відгука зменшена; - кнопка «подобається» не підсвічена червоним; - якщо оцінок більше немає, функціонального елемента «подобається» не має.
Результат тесту	Пройшов.

Таблиця 5.5 – Тестовий випадок «Створення груп гарячих ліній »

Функція	Створення груп гарячих ліній
Результат функції	<ul style="list-style-type: none"> - створено дві гарячі лінії з однаковими назвами; - об'єднання в одну лінію з двома номерами.
Результат тесту	Пройшов.

5.3 Інтеграційне тестування

Інтеграційне тестування - це тип тестування, щоб перевірити, чи працюють різні частини модулів разом.[14]

Поведінка інтеграційних модулів враховується при тестуванні інтеграції. Область тестування на інтеграцію широка, вона охоплює всю тестовану програму і вимагає значно більше зусиль, щоб зібрати разом.

Інтеграційне тестування залежить від інших зовнішніх систем, таких як бази даних, обладнання, виділене для них тощо. Цей тип тестування проводиться після тестування підрозділу. Тестування інтеграції починається із специфікації інтерфейсу. Інтеграційне тестування повинно проводитися для виявлення проблем, що виникають, коли різні модулі взаємодіють один з одним для створення загальної системи.

Метою інтеграційного тестування є тестування комбінованих модулів разом та гарантування того, що всі комбіновані модулі працюють як слід. Кожному тесту повинен відповідати окремий сценарій.

Всі опрацьовані сценарії, тестові випадки та їх результати є наведеними у таблицях 5.6-5.10.

Таблиця 5.6 – Тестовий випадок «Створення компанії»

Функція	Створення компанії
Результат функції	<ul style="list-style-type: none"> - компанія додана до бази даних; - користувачу надано признак адміністратора компанії; - користувача переадресувало на сторінку інформації про компанію.
Результат тесту	Пройшов.

Таблиця 5.7 – Тестовий випадок «Створити запитання»

Функція	Створити запитання
Результат функції	<ul style="list-style-type: none"> - запитання додано до бази даних; - запитанню надано ідентифікатор компанії; - якщо запитання надано за темою на ку призначений відповідальний, відповідальний отримує нотифікацію на пошту; - відкрито діалогове вікно з текстом «Дякую за ваше запитання, коли на нього дадуть відповідь ви отримаєте повідомлення на пошту »;
Результат тесту	Пройшов.

Таблиця 5.8 – Тестовий випадок «Відповідь на запитання»

Функція	Відповідь на запитання
Результат функції	<ul style="list-style-type: none"> - відповідь додано до бази даних; - відповіді надано ідентифікатор запитання; - користувачу відправлено повідомлення на пошту, що на його запитання надали відповідь; - запитання змінило статус на «Закрите»; - відображається список запитань;
Результат тесту	Пройшов.

Таблиця 5.9 – Тестовий випадок «Відправити запитання модератору»

Функція	Відправити запитання модератору
Результат функції	<ul style="list-style-type: none"> - користувач є відповідальним; - запитанню надано признак «відправлене модератору»; - в запитанні вказана причина відправлення; - модератору відправлено повідомлення на пошту; - відображення списку запитань без відправленого запитання.
Результат тесту	Пройшов.

Таблиця 5.10 – Тестовий випадок «Відправити відповідальному»

Функція	Призначити запитання відповідальному
Результат функції	<ul style="list-style-type: none"> - користувач є модератор; - запитанню надано признак «призначити»; - в запитанні вказана причина відправлення;

	<ul style="list-style-type: none"> - відповідальному відправлено повідомлення на пошту; - відображення списку запитань з відправленим запитанням.
Результат тесту	Пройшов.

5.4 End-to-end тестування

End-to-end тестування - тестує весь програмний продукт від початку до кінця, щоб переконатися, що потік додатків ведеться так, як очікувалося.[15]

Іншими словами, це допомагає визначити залежності продуктової системи та забезпечує взаємодію всіх інтегрованих деталей разом, як очікувалося. Кожному тесту повинен відповідати окремий сценарій. Всі опрацьовані сценарії, тестові випадки та їх результати є наведеними у таблицях 5.11-5.15.

Таблиця 5.11 – Тестовий випадок «Відповідь на відгук»

Функція	Залишити відповідь на відгук
Результат функції	<ul style="list-style-type: none"> - до відгуку добавлено дочірній елемент відгук - якщо відповідей більше одного, з'являється функціональний елемент «приховати відгуки»;
Результат тесту	Пройшов.

Таблиця 5.12 – Тестовий випадок «Відобразити відповіді»

Функція	Відобразити відгуки
Результат функції	<ul style="list-style-type: none"> - У відгука є більше одної відповіді; - Відображається один відгук та функціональний елемент «Відобразити всі відгуки»;

	- При натисканні на функціональний елемент «Відобразити всі відгуки» відображається більше одного відгука;
Результат тесту	Пройшов.

Таблиця 5.13 – Тестовий випадок «Видалення відгука»

Функція	Видалення відгука
Результат функції	- відображається діалогове вікно з підтвердженням; - при підтвердженні дії елемент відгуку видаляється зі сторінки.
Результат тесту	Пройшов.

Таблиця 5.14 – Тестовий випадок «Валідація форми реєстрації»

Функція	Валідація форми реєстрації
Результат функції	- введені невірні дані; - відображається текст повідомлення про некоректність даних; - кнопка «zareestruvatisia» неактивна.
Результат тесту	Пройшов.

Таблиця 5.15 – Тестовий випадок «Видалення компанії»

Функція	Видалення компанії
Результат функції	- відображається діалогове вікно з підтвердженням;

	- при підтвердженні дії переправлено на сторінку перегляду списків компаній.
Результат тесту	Пройшов.

Висновки до розділу

В даному розділі було розроблено автоматичне тестування для проекту. Було використано три способи тестування :

- unit-тестування;
- інтеграційне тестування;
- тестування «end-to-end».

Завдяки тестування різними способами були перевірені різноманітні аспекти продукту та було передбачено та виправлено велика кількість недоліків. Можна бути впевненим, що була досягнена хороша якість продукту.

					IT61.140БАК.004 ПЗ	Арк.
Змн.	Арк.	№ докум.	Підпис	Дата		90

6 ВПРОВАДЖЕННЯ ТА ВИКОРИСТАННЯ СИСТЕМИ

6.1 Апаратні вимоги для використання системи

Документація для запуску програми є важливою частиною проекту. Вона допоможе за декілька хвилин встановити програму на окреме середовище, для того щоб додати нові функціональні можливості або виправити недоліки які будуть виявлені в процесі використання. Це дасть змогу новим розробникам більш зрозуміти головні аспекти програми і допоможе в подальшій реалізації системи.

Було розроблено автоматизовану систему зворотного зв'язку на базі кросплатформових фреймворків Angular та .Net core, що забезпечує підтримку різноманітних операційних систем при невеликому навантаженні на сервер

Мінімальні системні вимоги до сервера для розробки описані у таблиці 6.1

Таблиця 6.1 – мінімальні системні вимоги до апаратної частини сервера

Процесор	Процесор з двома ядрами або більше та з тактовою частотою більше 1ГГц.
Місце на диску	Більше 10 гігабайт.
Оперативна пам'ять	8 гігабайт.
Операційні системи	Linux, Windows XP/7/8/10, macOS.

6.2 Інструкція з встановлення для розробки

Для розробки цього проекту потрібно встановити компілятор Angular CLI для запуску клієнтської частини, а для підняття серверу потрібно встановити пакет SDK Microsoft, який в свою чергу включає можливість запуску бази даних Microsoft Server SQL.

При запуску програми потрібно слідувати наступним інструкціям:

- 1) перейти до кореневої папки серверної частини в консолі;
- 2) запустити серверну частину за допомогою команди `dotnet run`;
- 3) перейти до кореневої папки клієнтської частини частини в консолі;
- 4) встановити допоміжні пакети за допомогою команди `npm i`;
- 5) запустити клієнтську частину за допомогою команди `ng serve`.

6.3 Висновки до розділу

Було визначено рекомендовані системні вимоги до апаратної частини сервера. Створено інструкцію з встановлення для розробки.

					IT61.140БАК.004 ПЗ	Арк.
						92
Змн.	Арк.	№ докум.	Підпис	Дата		

ВИСНОВКИ

В дипломному проєкті була розроблена система автоматизованого зворотного зв'язку. В процесі реалізації було проведено:

- аналіз існуючих рішень, які дали змогу уникнути вже існуючих проблем та зрозуміти доцільність створення нової системи;
- аналіз особливостей систем я яких впливали функціональні та нефункціональні вимоги системи;
- аналіз необхідної ролевої моделі та створення сценаріїв використання відносно різних ролей;
- аналіз існуючих технологій розробки, які були вибрані відносно функціональних та нефункціональних вимог системи;
- створення прототипу системи, без дизайну, для подальшого розуміння розробки системи;
- створення структурної схеми проєкту для розуміння архітектури проєкту, було вибрано архітектурний шаблон model-view-controller;
- створення архітектури бази даних з передбаченням можливості масштабування системи та можливістю оброблювання даних;
- створення серверної сторони, для обробки запитів, відправлення нотифікацій;
- створення інтерфейсу користувача, для інтуїтивного управління системою;
- тестування всієї системи, для досягнення хорошої якості продукту.

Створений програмний продукт є готовим до використання та буде з часом розширений новими модулями зворотного зв'язку.

					IT61.140БАК.004 ПЗ	Арк.
						93
Змн.	Арк.	№ докум.	Підпис	Дата		

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. The Royal Gazette online news, business, sports, Island life, jobs and newsletters. URL: <http://www.royalgazette.com/article/20130507/COLUMN21/705079979> (дата звернення: 01.05.2020).
2. An American commercial infotainment website. URL: <https://computer.howstuffworks.com/internet/social-networking/networks/yelp2.html> (дата звернення: 09.05.2020).
3. Товариство відгуків про все на світі URL: <https://otzovik.com/> (дата звернення: 09.05.2020).
4. Товариство програмістів URL: <https://dou.ua/> (дата звернення: 09.05.2020).
5. Showcasing the best websites. URL: <https://www.designbombs.com/siteground-review/> (дата звернення: 09.05.2020).
6. The customer knowledge blog URL: <https://www.myfeedback.com/en/blog/top-brands-collect-customer-feedback> (дата звернення: 09.05.2020).
7. The free encyclopedia. URL: https://en.wikipedia.org/wiki/Google_Maps (дата звернення: 09.05.2020).
8. An American developer and marketer of software products for inbound marketing, sales, and customer service. URL: <https://blog.hubspot.com/service/state-of-service-2019-customer-first> (дата звернення: 13.05.2020).
9. Адам Фримен Angular для професіоналов, 2017 - 800 с.
10. Адам Фримен ASP.NET MVC 5 с прикладами на C# 5.0 для професіоналов, 2017. - 736 с.
11. Borisyuzhakov Пять простых шагов для понимания JSON Web Tokens (JWT) URL: <https://habr.com/ru/post/340146/> (дата звернення: 14.05.2020).
12. Vkhorkov, DTO vs POCO vs Value Object. URL: <https://habr.com/ru/post/268371/> (дата звернення: 14.05.2020).
13. Marshinov, Юніт-тестування URL: <https://habr.com/ru/post/169381/> (дата звернення: 01.06.2020).

					ІТ61.140БАК.004 ПЗ	Арк.
						94
Змн.	Арк.	№ докум.	Підпис	Дата		

14. Rami Honig, 6 best practices for integration testing with continuous integration URL:<https://techbeacon.com/devops/6-best-practices-integration-testing-continuous-integration> (дата звернення: 01.06.2020).

15. Wwakabobik, (end-to-end): що, нащо, чому URL:
<https://habr.com/ru/post/417395/> (дата звернення: 02.06.2014).

					IT61.140БАК.004 ПЗ	Арк.
						95
Змн.	Арк.	№ докум.	Підпис	Дата		